Vicenç Torra
Yasuo Narukawa
Yuji Yoshida (Eds.)

LNAI 4617

Modeling Decisions for Artificial Intelligence

4th International Conference, MDAI 2007
Kitakyushu, Japan, August 2007
Proceedings

Springer

Lecture Notes in Artificial Intelligence 4617

Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Vicenç Torra Yasuo Narukawa
Yuji Yoshida (Eds.)

Modeling Decisions for Artificial Intelligence

4th International Conference, MDAI 2007
Kitakyushu, Japan, August 16-18, 2007
Proceedings

Springer

Series Editors

Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Volume Editors

Vicenç Torra
IIIA-CSIC
Campus UAB s/n, 08193 Bellaterra, Catalonia, Spain
E-mail: vtorra@iiia.csic.es

Yasuo Narukawa
Toho Gakuen
3-1-10 Naka, Kunitachi, Tokyo, 186-0004, Japan
E-mail: narukawa@d4.dion.ne.jp

Yuji Yoshida
University of Kitakyushu
Faculty of Economics and Business Administration
4-2-1 Kitagata, Kokuraminami, Kitakyushu 802-8577, Japan
E-mail: yoshida@kitakyu-u.ac.jp

Library of Congress Control Number: 2007931447

CR Subject Classification (1998): I.2, F.4.1, F.1, H.2.8, I.6

LNCS Sublibrary: SL 7 – Artificial Intelligence

ISSN 0302-9743
ISBN-10 3-540-73728-6 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-73728-5 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12093700 06/3180 5 4 3 2 1 0

Preface

This volume contains papers presented at the 4th International Conference on Modeling Decisions for Artificial Intelligence (MDAI 2007), held in Kitakyushu, Japan, August 16-18. This conference followed MDAI 2004 (Barcelona, Catalonia, Spain), MDAI 2005 (Tsukuba, Japan), and MDAI 2007 (Tarragona, Catalonia, Spain) with proceedings also published in the LNAI series (Vols. 3131, 3558, and 3885).

The aim of this conference was to provide a forum for researchers to discuss the theory and tools for modeling decisions as well as applications that encompass decision-making processes and information-fusion techniques.

The organizers received 193 papers from 21 different countries, from Asia, Europe, America, Africa, and Australia, 42 of which are published in this volume. Each submission received at least two reviews from the Program Committee and a few external reviewers. We would like to express our gratitude to them for their work. The plenary talks presented at the conference are also included in this volume.

The conference was supported by the University of Kitakyushu, the UNESCO Chair in Data Privacy, the Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT), the Catalan Association for Artificial Intelligence (ACIA), the European Society for Fuzzy Logic and Technology (EUSFLAT), and the City of Kitakyushu.

May 2007

Vicenç Torra
Yasuo Narukawa
Yuji Yoshida

Modeling Decisions for Artificial Intelligence – MDAI 2007

General Chair

Yuji Yoshida, University of Kitakyushu, Japan

Program Chairs

Vicenç Torra, IIIA-CSIC, Catalonia, Spain
Yasuo Narukawa, Toho Gakuen, Japan

Program Committee

G. Beliakov (Australia)
U. Bodenhofer (Austria)
J. Domingo-Ferrer (Spain)
J. Dujmovic (USA)
B. H. Far (Canada)
M. Grabisch (France)
E. Herrera-Viedma (Spain)
K. Hirota (Japan)
M. Inuiguchi (Japan)
J. Kacprzyk (Poland)
J.-L. Marichal (Luxemburg)
R. Meo (Italy)
R. Mesiar (Slovak Republic)
S. Miyamoto (Japan)
T. Murofushi (Japan)
M. Ng (Hong Kong, China)
T. Onisawa (Japan)
G. Pasi (Italy)
C. Sierra (Spain)
R. R. Yager (USA)
N. Zhong (Japan)

Local Organizing Committee Chairs

Satoru Kumamoto, University of Kitakyushu, Japan
Eiju Hirowatari, University of Kitakyushu, Japan

Additional Referees

J. Sabater, S. Igarashi, J. L. Arcos, J. Herranz, Y. Hayashi, T. Furuhashi, M. Ogiela, J. Nin, Y. Maeda, K. Fujimoto, M. Hagiwara, A. Solanas, J. Castella-Roca, V. Daza, A. Viejo, F. Sebé, A. Martinez-Ballesté, G. Escalada-Imaz, J. Ozawa, D. Nettleton, Y. Kimura, S. Saito, I. Kojadinovic.

Supporting Institutions

The University of Kitakyushu
The UNESCO Chair in Data Privacy
The Japan Society for Fuzzy Theory and Intelligent Informatics (SOFT)
The Catalan Association for Artificial Intelligence (ACIA)
The European Society for Fuzzy Logic and Technology (EUSFLAT)
The City of Kitakyushu

Table of Contents

Invited Papers

Decision Making

Non Additive Measures and Concept Lattices

Clustering and Rough Sets

Soft Computing

Applications

An Overview of Fuzzy Relational Calculus and Its Applications

Etienne E. Kerre

Fuzziness & Uncertainty Modelling,
Department of Applied Mathematics & Computer Science,
Ghent University, Krijgslaan 281 - S9,
B-9000, Belgium
etienne.kerre@ugent.be
http://www.fuzzy.ugent.be

The calculus of relations has been very important during the past 40 years from theoretical as well as from practical point of view. The development of fuzzy set theory, particularly in the framework of relational calculus has undoubtly increased the interest in this domain of science.

In this paper I will give a brief overview of the most recent colourings of the classical theory of relations and convince the reader about the high applicability of fuzzy relational calculus in providing examples from information retrieval, relational databases, approximate reasoning, preference modelling, medical diagnosis. Most of the work presented here has been developed in my research team at Ghent University during the past 30 years.

1 New Concepts in Classical Relational Calculus

The concept of a relation is fundamental since only in a few steps one can introduce this concept in the framework of set theory. Indeed as soon as the meaning of the so-called classifier $\{z|P\}$, i.e., the class of all objects z that satisfy a given property P, has been introduced (intuitively in the sense of Cantor or axiomatically in the sense of Gödel-Bernays-Von Neumann or Zermelo-Fraenkel) one may define a singleton, a doubleton, an ordered pair, the cartesian product of two sets and finally a relation from X to Y as a subset of the cartesian product $X \times Y$. This concept may be extended to a relation between n universes $X_1, X_2, \ldots, X_n$ as a subset of $X_1 \times X_2 \times \ldots \times X_n$. We all are familiar with a special kind of relations, namely the functional relations or shortly functions. It is hard to image mathematics without the concept of a relation, in particular without the concept of a function. Some auxiliary notions with respect to a relation R from X to Y are:

- The domain $\mathrm{dom}(R)$ consisting of all elements of X that are coupled by R to at least one element of Y.
- The range $\mathrm{rng}(R)$ consisting of all elements of Y that are linked to at least one element of X.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 1–13, 2007.

- The inverse R^{-1} consisting of all ordered pairs (y, x) such that (x, y) belongs to R.
- The R-afterset of x (denoted xR) consisting of all elements of Y that are linked to x.
- The R-foreset of y (denoted Ry) consisting of all elements of X that are linked to y.

Due to the last two notions introduced by Bandler and Kohout [1] in the 80's, a lot of new concepts concerning images and compositions could be introduced. These concepts substantially enlarge the toolkit of relational calculus.

Since relations are sets all set-theoretic operations such as union, intersection, complementation, difference, symmetric difference as well as the natural join operation can be applied to relations. For example suppose that R_1 and R_2 are relations from X to Y, then the union $R_1 \cup R_2$ consists of all ordered pairs (x, y) that belong to R_1 or to R_2. It is interesting to note that all these operations can be directly applied to aftersets and foresets, i.e., the $(R_1 \cup R_2)$-afterset of $x \in X$ equals the union of xR_1 and xR_2, i.e., the family of aftersets $(xR)_{x \in X}$ contains all relevant information concerning the relation R and similarly for the family of foresets $(Ry)_{y \in Y}$.

Important notions in mathematics like continuity and measurability are based on the concept of direct and inverse image of a set under a (functional) relation.

Let R be a relation from X to Y, A a subset of X and B a subset of Y, then

- the direct image of A under R is given by:

$$R(A) = \{y | (\exists x \in A)((x, y) \in R)\}$$

  or equivalently using after- and foresets:

$$\begin{aligned} R(A) &= \{y | A \cap Ry \neq \emptyset\} \\ &= \bigcup_{x \in A} xR \end{aligned}$$

- the inverse image of B under R is given by:

$$\begin{aligned} R^{-1}(B) &= \{x | (\exists y \in B)((x, y) \in R)\} \\ &= \{x | B \cap xR \neq \emptyset\} \\ &= \bigcup_{y \in B} Ry \end{aligned}$$

Inspired by the work of Bandler-Kohout [1] on the new compositions, De Baets-Kerre [2-5] have introduced some new images that could be defined using after- and foresets:

- the subdirect image of A under R:

$$\begin{aligned} R^{\triangleleft}(A) &= \{y | A \cap Ry \neq \emptyset \quad and \quad A \subseteq Ry\} \\ &= \{y | A \neq \emptyset \quad and \quad Ry \neq \emptyset \quad and \quad A \subseteq Ry\} \end{aligned}$$

– the superdirect image of A under R:

$$\begin{aligned} R^{\triangleright}(A) &= \{y | A \cap Ry \neq \emptyset \quad and \quad Ry \subseteq A\} \\ &= \{y | A \neq \emptyset \quad and \quad Ry \neq \emptyset \quad and \quad Ry \subseteq A\} \end{aligned}$$

– the squaredirect image of A under R:

$$\begin{aligned} R^{\square}(A) &= \{y | A \cap Ry \neq \emptyset \quad and \quad A = Ry\} \\ &= \{y | A \neq \emptyset \quad and \quad Ry \neq \emptyset \quad and \quad A = Ry\} \end{aligned}$$

Finally we mention the most important operation on relations, namely composition or product and its useful extensions introduced by Bandler-Kohout [1] and slightly modified by De Baets-Kerre [2-5].

Let R_1 be a relation from X to Y and R_2 a relation from Y to Z, then:

– the round product of R_1 and R_2 (read R_1 before R_2, R_1 followed by R_2) is defined as the relation from X to Z given by:

$$R_1 \circ R_2 = \{(x, z) | (\exists y)((x, y) \in R_1 \quad and \quad (y, z) \in R_2)\}$$

or equivalently using after- and foresets:

$$R_1 \circ R_2 = \{(x, z) | xR_1 \cap R_2 z \neq \emptyset\}$$

– the subproduct of R_1 and R_2:

$$\begin{aligned} R_1 \triangleleft R_2 &= \{(x, z) | xR_1 \cap R_2 z \neq \emptyset \quad and \quad xR_1 \subseteq R_2 z\} \\ &= \{(x, z) | xR_1 \neq \emptyset \quad and \quad R_2 z \neq \emptyset \quad and \quad xR_1 \subseteq R_2 z\} \end{aligned}$$

– the superproduct of R_1 and R_2:

$$\begin{aligned} R_1 \triangleright R_2 &= \{(x, z) | xR_1 \cap R_2 z \neq \emptyset \quad and \quad R_2 z \subseteq xR_1\} \\ &= \{(x, z) | xR_1 \neq \emptyset \quad and \quad R_2 z \neq \emptyset \quad and \quad R_2 z \subseteq xR_1\} \end{aligned}$$

– the squareproduct of R_1 and R_2:

$$\begin{aligned} R_1 \square R_2 &= \{(x, z) | xR_1 \cap R_2 z \neq \emptyset \quad and \quad xR_1 = R_2 z\} \\ &= \{(x, z) | xR_1 \neq \emptyset \quad and \quad R_2 z \neq \emptyset \quad and \quad xR_1 = R_2 z\} \end{aligned}$$

2 A Brief Outline of Fuzzy Relational Calculus

Since the old Greeks scientists have recognized that binary or black-or-white logic is not sufficient to model our knowledge which is mostly pervaded with imprecision. We have to wait until 1965 when Lotfi Zadeh introduced the concept of a fuzzy set in his seminal paper entitled "Fuzzy Sets",in order to model imprecise terms as "sets" with unsharp bounderies where the transition from belonging to not belonging is rather gradual than abrupt. In the same spirit Zadeh introduced the concept of a fuzzy relation from a universe X to a universe Y as

a fuzzy set R in the cartesian product $X \times Y$ where $R(x, y)$ denotes the strength of relationship between $x \in X$ and $y \in Y$.

More formally a fuzzy relation R from X to Y is a mapping from $X \times Y$ into the unit interval $[0, 1]$, attaching to every ordered pair (x, y) in $X \times Y$ a degree of relationship $R(x, y)$ belonging to $[0, 1]$.

The basic concepts introduced in section 1 can be generalized or fuzzified as follows. Let R be a fuzzy relation from X to Y, then:

- the domain of R is a fuzzy set in X given by:

$$\mathrm{dom}(R)(x) = \sup\{R(x, y) | y \in Y\}, \forall x \in X$$

- the range of R is a fuzzy set in Y given by:

$$\mathrm{rng}(R)(y) = \sup\{R(x, y) | x \in X\}, \forall y \in Y$$

- the inverse R^{-1} of R is the fuzzy relation from Y to X given by:

$$R^{-1}(y, x) = R(x, y), \forall (y, x) \in Y \times X$$

- the R-afterset of $x \in X$ is the fuzzy set in Y given by:

$$xR(y) = R(x, y), \forall y \in Y$$

- the R-foreset of $y \in Y$ is the fuzzy set in X given by:

$$Ry(x) = R(x, y), \forall x \in X$$

All the set-theoretic operations have been extended in an infinite number of ways to fuzzy sets and a fortiori to fuzzy relations using the concepts of triangular norms and conorms introduced by Schweizer-Sklar in the framework of probabilistic metric spaces. A triangular norm T is a $[0, 1]^2 - [0, 1]$ mapping satisfying commutativity, associativity, monotonicity and the boundary condition $T(x, 1) = x, \forall x \in [0, 1]$. A triangular conorm S is defined in a similar way but with the boundary condition $S(x, 0) = x, \forall x \in [0, 1]$. The T-intersection (S-union) of two fuzzy relations R_1 and R_2 from X to Y is defined as a fuzzy relation from X to Y given as:

$$R_1 \cap_T R_2(x, y) = T(R_1(x, y), R_2(x, y))$$

$$R_1 \cup_S R_2(x, y) = S(R_1(x, y), R_2(x, y))$$

for all $(x, y) \in X \times Y$.

All operations on fuzzy relations may be defined using after- and foresets, i.e.:

$$x(R_1 \cap_T R_2) = xR_1 \cap_T xR_2, \forall x \in X$$

$$(R_1 \cup_S R_2)y = R_1 y \cup_S R_2 y, \forall y \in Y$$

In order to fuzzify the concepts of images and compositions we need an extension of the classical intersection or conjunction operation and the binary

implication to model the inclusion. As said before the intersection of two fuzzy sets in some universe may be modelled by a triangular norm ranging from the smallest one (the drastic one) to the largest one being the minimum operator. In this way we obtain for the T-direct image of a fuzzy set A in X under a fuzzy relation R from X to Y:

$$\begin{aligned} R_T(A) : Y &\to [0,1] \\ y &\mapsto \sup_{x \in X} T(A(x), R(x,y)), \forall y \in Y \end{aligned}$$

Similarly for the T-round composition of a fuzzy relation R_1 from X to Y followed by a fuzzy relation R_2 from Y to Z we obtain:

$$\begin{aligned} R_1 \circ_T R_2 : X \times Z &\to [0,1] \\ (x,z) &\mapsto \sup_{y \in Y} T(R_1(x,y), R_2(y,z)), \forall (x,z) \in X \times Z \end{aligned}$$

A fuzzy implication is defined as a $[0,1]^2 - [0,1]$ mapping I satisfying the boundary conditions: $I(0,0) = I(0,1) = I(1,1)$ and $I(1,0) = 0$.

Putting extra conditions such as hybrid monotonicity, neutrality principle and exchange principle leads to more specific implication operators. Some popular operators are:

- the Kleene-Dienes implication I_{KD} given as:

$$I_{KD}(x,y) = \max(1-x, y)$$

- the Lukasiewicz implication I_L given as:

$$I_L(x,y) = \min(1, 1-x+y)$$

- the Reichenbach implication I_R given as:

$$I_R(x,y) = 1 - x + xy$$

Let R be a fuzzy relation from X to Y, A a fuzzy set in X, T a triangular norm and I a fuzzy implication. Then we define:

- the T-I subdirect image of A under R as:

$$\begin{aligned} R^{\triangleleft}_{T,I}(A) : Y &\to [0,1] \\ y &\mapsto \min(\sup_{x \in X} T(A(x), R(x,y)), \inf_{x \in X} I(A(x), R(x,y))), \forall y \in Y \end{aligned}$$

- the T-I superdirect image of A under R as:

$$\begin{aligned} R^{\triangleright}_{T,I}(A) : Y &\to [0,1] \\ y &\mapsto \min(\sup_{x \in X} T(A(x), R(x,y)), \inf_{x \in X} I(R(x,y), A(x))), \forall y \in Y \end{aligned}$$

- the T-I squaredirect image of A under R as:

$$\begin{aligned} R^{\square}_{T,I}(A) : Y &\to [0,1] \\ y &\mapsto \min(R^{\triangleleft}_{T,I}(A)(y), R^{\triangleright}_{T,I}(A)(y)), \forall y \in Y \end{aligned}$$

Finally let us fuzzify the new compositions of fuzzy relations. Let R_1 be a fuzzy relation from X to Y, R_2 a fuzzy relation from Y to Z, T a triangular norm and I a fuzzy implication. Then we define:

- the T-I subproduct of R_1 and R_2 as:

$$R_1 \triangleleft_{T,I} R_2 : X \times Z \to [0,1]$$
$$(x,z) \mapsto \min(\sup_{y \in Y} T(R_1(x,y), R_2(y,z)), \inf_{y \in Y} I(R_1(x,y), R_2(y,z))),$$
$$\forall (x,z) \in X \times Z$$

- the T-I superproduct of R_1 and R_2 as:

$$R_1 \triangleright_{T,I} R_2 : X \times Z \to [0,1]$$
$$(x,z) \mapsto \min(\sup_{y \in Y} T(R_1(x,y), R_2(y,z)), \inf_{y \in Y} I(R_2(y,z), R_1(x,y))),$$
$$\forall (x,z) \in X \times Z$$

- the T-I squareproduct of R_1 and R_2 as:

$$R_1 \square_{T,I} R_2 : X \times Z \to [0,1]$$
$$(x,z) \mapsto \min(R_1 \triangleleft_{T,I} R_2(x,z), R_1 \triangleright_{T,I} R_2(x,z)), \forall (x,z) \in X \times Z$$

For more detailed information about the basics of fuzzy relational calculus and some of its extensions such as to intuitionistic fuzzy set theory and rough set theory we refer to [6-16].

3 A List of Successful Applications of Fuzzy Relational Calculus

In this section I want to convince the reader about the huge number of domains where fuzzy relations can be applied. Once more I will restrict myself to those domains in which my research team has been active during the past 30 years.

3.1 Information Retrieval

In a document retrieval system there are basically 3 finite sets involved: a set $U = \{u_1, \ldots, u_p\}$ of users of the system, a set $D = \{d_1, \ldots, d_q\}$ of documents constituting the system and a set $T = \{t_1, \ldots, t_r\}$ of terms (descriptors, keywords) describing the documents. The document retrieval system can be described by means of a fuzzy relation F from D to T where $F(d,t)$ indicates for every pair (d,t) belonging to $D \times T$ to what degree a document d deals with a term t. From such a document description relation a lot of information can be obtained using the apparatus of fuzzy relational calculus. Some examples:

- the F-afterset of a document d gives the fuzzy set of terms that are treated in this document,
- the F-foreset of a term t gives the fuzzy set of documents that deal with t,

- the compositions $F \circ_T F^{-1}$, $F \triangleleft_{T,I} F^{-1}$, $F \triangleright_{T,I} F^{-1}$, $F \square_{T,I} F^{-1}$ are fuzzy relations from D to D that measure the overlap or redundancy of information available in the documents,
- the compositions $F^{-1} \circ_T F$, $F^{-1} \triangleleft_{T,I} F$, $F^{-1} \triangleright_{T,I} F$, $F^{-1} \square_{T,I} F$ are fuzzy relations from T to T that measure the dependencies between the terms,
- the compositions $G \circ_T F^{-1}$, $G \triangleleft_{T,I} F^{-1}$, $G \triangleright_{T,I} F^{-1}$, $G \square_{T,I} F^{-1}$, where G is a fuzzy relation from the set U of users to the set T of terms, are fuzzy relations from U to D that measure the interest of the users in the document available.

For more information we refer to [17-21].

3.2 Relational Databases

Relational calculus has started its development with the introduction of set theory by Cantor at the end of the 19th century. However we had to wait till 1970 when Codd launched the most popular database management system based on relational calculus for a renewed interest in this domain of mathematics. A classical relational data model consists in a finite number of finite sets $D_1, \ldots, D_n$ called domains and a number of variables $A_1, \ldots, A_n$ called attributes, that assume values in the respective sets $D_1, \ldots, D_n$. Now the data can be represented by means of n-ary relations between $D_1, \ldots, D_n$.

The class of all n-ary relations on $D_1, \ldots, D_n$ is called the relational scheme on the domains $D_1, D_2, \ldots, D_n$. Then relational algebra has been used to provide the answer to the queries. Like other classical models, the classical relational data model is also of an all-or-nothing nature in data representation as welll as in data manipulation, i.e., anything in or about the database should be precisely defined (for example: salary = 3.000 EUR per month) and represented. That's why the creation of fuzzy relational databases in which attribute values may be imprecise (salary=between 2.500 and 3.500 EUR per month) or fuzzy (salary = high) and where queries may be formulated in natural language (necessarily containing vague predicates), has substantially enhanced the applicability of relational databases. For more details about important concepts such as data redundancy, functional dependencies, normal forms and integrity we refer to [22-40].

3.3 Application to Approximate Reasoning

Commonsense knowledge in general and expert knowledge in particular are pervaded with imprecision and uncertainty. Relational calculus is very useful to treat both aspects of incomplete information. The knowledge base of an expert system consists of a database of facts (for example: unemployment is very high) and a rulebase of so-called IF-THEN rules (for example: if unemployment is high then salaries will be low) being of the general form: IF X is A THEN Y is B, where: X is a variable taking values in a universe U,
Y is a variable taking values in a universe V,
A is a fuzzy set in U,
B is a fuzzy set in V.

Such a rule can be modelled by means of a fuzzy relation R from U to V: $R(u,v) = I(A(u), B(v))$ where I represents a fuzzy implication, i.e., a $[0,1]^2 - [0,1]$ mapping satisfying the boundary conditions $I(0,0) = I(0,1) = I(1,1) = 1$ and $I(1,0) = 0$. Now suppose we also know that "X is A'" where A' is a fuzzy set in U. Then we can infer (the so-called generalized modus ponens) a restriction on the variable $Y : Y$ is B' where $B' = R_T(A')$.

This compositional rule of inference can be generalized to an arbitrary number of relational connections between an arbitrary number of variables using the concepts of cylindrical extension and projection. For more details see [44-45].

Another important application of fuzzy relations, in particular resemblance relations, in approximate reasoning concerns the representation of linguistic hedges (more-or-less, very, roughly) used for the modification of linguistic values. For more details we refer to [50].

The concept of a rough set as introduced by Z. Pawlak is based on an approximation space, i.e., a non-empty universe together with an equivalence relation. Generalizing an equivalence relation to a similarity relation R (i.e., a fuzzy relation satisfying relexivity, symmetry and sup-min transitivity) leads to a so-called fuzzy approximation space (see [55]). For every fuzzy set A in X, a fuzzy implicator I and a triangular norm T, we defined a lower and an upper fuzzy rough approximation of A as:

$$\underline{R}_I(A)(x) = \inf_{y \in X} I(R(x,y), A(y)), \forall x \in X$$

$$\overline{R}_T(A)(x) = \sup_{y \in X} T(R(x,y), A(y)), \forall x \in X$$

i.e., the lower fuzzy rough approximation is closely related to the T-I superdirect image of A under R, while the upper fuzzy rough approximation equals the T-direct image of A under R. For more information about fuzzy rough sets consisting of an ordered pair of a lower and an upper fuzzy rough approximation we refer to [55]. In [54] we gave several characterizations of important classes of (L-) fuzzy relations (serial, reflexive, irreflexive, symmetric, L-transitive, L-cotransitive, L-Euclidean and L-coEuclidean) in terms of fuzzy modal operators.

Further applications of fuzzy relational calculus for approximate reasoning can be found in [41-57].

3.4 Application to Preference Modelling

In a decision making problem one is usually confronted with a set A of alternatives among which the best one has to be choosen. The decision maker has to compare any two alternatives a and b and has to choose one of the following statements: a is preferred to b, b is preferred to a, indifferent to a and b, unable to compare a and b. In this way 3 crisp binary relations on A are defined: P (preference), I (indifference) and J (incomparability), satisfying:

(1) I is reflexive and J is irreflexive,
(2) P is asymmetric,

(3) I and J are symmetric,
(4) $P \cap I = \emptyset$ and $P \cap J = \emptyset$ and $I \cap J = \emptyset$
(5) $P \cup P^t \cup I \cup J = A^2$,

where P^t denotes the transpose of P.

In a series of papers we have investigated the fuzzified version of such a classical preference structure. For more details we refer to [71-77].

3.5 Application to Image Processing, Ordering Techniques and Medical Diagnosis

Because of space limitation I cannot provide more details about these applications. Application to image processing techniques, especially to fuzzy mathematical morphology can be found in [58-62]. The use of fuzzy relational calculus for fuzzy order relations and for ranking fuzzy quantities has been demonstrated in [63-67]. Finally some applications of fuzzy relations to medical diagnosis have been described in [68-70].

References

1. Bandler, W., Kohout, L.: Fuzzy relational products as a tool for analysis and synthesis of the behavior of complex natural and artificial systems. In: Wang, P.P., Chang, S.K. (eds.) Fuzzy Sets Theory and Applications to Policy Analysis and Information Systems, pp. 341–367. Plenum Press, New York (1980)
2. De Baets, B., Kerre, E.E.: Triangular fuzzy relational compositions revisited. In: Bouchon, B., Valverde, L., Yager, R. (eds.) Uncertainty in Intelligent Systems, pp. 257–267. Elsevier, North Holland (1993)
3. De Baets, B., Kerre, E.E.: Some reflections on fuzzy relational compositions. In: Valverde, L., Bouchon-Meunier, B., Yager, R.R. (eds.) IPMU 1992. LNCS, vol. 682, pp. 251–254. Springer, Heidelberg (1993)
4. De Baets, B., Kerre, E.E.: A revision of Bandler-Kohout compositions of relations. Mathematica Pannonica 4, 59–78 (1993)
5. De Baets, B., Kerre, E.E.: Fuzzy relational compositions. Fuzzy Sets and Systems 60, 109–120 (1993)
6. De Baets, B., Kerre, E.E.: On the supercomposition of subcontinuous and fuzzy subcontinuous mappings and related topics. The Journal of Fuzzy Mathematics 2, 17–34 (1994)
7. De Baets, B., Kerre, E.E.: The cutting of compositions. Fuzzy Sets and Systems 62, 295–309 (1994)
8. De Baets, B., Kerre, E.E.: A primer on solving fuzzy relational equations on the unit interval. Fuzziness and Knowledge-Based Systems 2, 205–225 (1994)
9. De Baets, B., Kerre, E.E.: Fuzzy relations and applications. In: Hawkes, P. (ed.) Advances in Electronics and Electron Physics, pp. 255–324. Academic Press, San Diego (1994)
10. Wang, X., De Baets, B., Kerre, E.E.: A comparative study of similarity measures. Fuzzy Sets and Systems 73(2), 259–268 (1995)
11. Nachtegael, M., De Cock, M., Vanderweken, D., Kerre, E.E.: Fuzzy relational images in computer science. In: de Swart, H. (ed.) RelMiCS 2001. LNCS, vol. 2561, pp. 134–151. Springer, Heidelberg (2002)

12. Cornelis, C., Deschrijver, G., De Cock, M., Kerre, E.E.: Intuitionistic fuzzy relational calculus: an overview. In: Proceedings of First International IEEE Symposium on Intelligent Systems, Varna, Bulgaria, vol. I, pp. 340–345. IEEE Computer Society Press, Los Alamitos (2002)
13. Radzikowska, A.-M., Kerre, E.E.: A fuzzy generalization of information relations. In: Fitting, M., Orlowska, E. (eds.) Beyond Two: Theory and Applications of Multiple-Valued Logic, pp. 287–312. Physica Verlag, Heidelberg (2003)
14. Deschrijver, G., Kerre, E.E.: On the composition of intuitionistic fuzzy relations. Fuzzy Sets and Systems 136(3), 333–362 (2003)
15. Deschrijver, G., Kerre, E.E.: On the cuts of intuitionistic fuzzy compositions. Kuwait Journal of Science and Engineering 32(1), 17–38 (2005)
16. De Cock, M., Cornelis, C., Kerre, E.E.: Intuitionistic fuzzy relational images. In: Halgamuge, S.K., Wang, L. (eds.) Computational Intelligence for Modeling and Predictions. Studies in Computational Intelligence, vol. 2, pp. 129–146. Springer, Heidelberg (2005)
17. Kerre, E.E., Zenner, R., De Caluwe, R.: The Use of Fuzzy set Theory in Information Retrieval and Databases: a Survey. Journal American Society for Information Science 37(5), 341–345 (1986)
18. Zenner, R., De Caluwe, R., Kerre, E.E.: A New Approach to Information Retrieval systems using Fuzzy Expressions. Fuzzy Sets and Systems 17, 9–22 (1985)
19. Kerre, E.E., De Cooman, G., Ruan, D., Cappelle, B., Chen, G.: Introduction to the Basic Principles of Fuzzy Set Theory and some of its Applications, second revised edition. Communication and Cognition, Blandijnberg 2, B-9000 Gent, p. 360 (1993) (ISBN 90-70963-41-8)
20. Kerre, E.E.: Fuzzy set theory in information retrieval and databases. In: Kent, A., Williams, J. (eds.) Encyclopedia of Computer Science and Technology, vol. 21(suppl. 6), pp. 117–138. Marcel Dekker Inc. New York (1990)
21. Kerre, E.E.: Fuzzy relational calculus in terms of after– and foresets and its application to information retrieval. In: AL-Hussaini, E.K., Abou–El–Ela, A.M. (eds.) Proceedings of Assiut First International Conference on Mathematics and Statistics, Part VIII, Assiut, pp. 133–157 (1990)
22. Vandenberghe, R., Van Schooten, A., De Caluwe, R., Kerre, E.E.: Some practical aspects of fuzzy database techniques: an example. Information Systems 14(6), 465–472 (1989)
23. Chen, G., Vandenbulcke, J., Kerre, E.E.: A step towards the theory of fuzzy relational database design. In: Lowen, B., Roubens, M. (eds.) Proceedings IFSA '91, Brussels, pp. 44–47 (1991)
24. Chen, G., Vandenbulcke, J., Kerre, E.E.: A general treatment of data redundancy in a fuzzy relational data model. Journal of the American Society for Information Science 43(4), 304–311 (1992)
25. Chen, G., Kerre, E.E., Vandenbulcke, J.: Fuzzy functional dependency and axiomatic system in a fuzzy relational data model. In: Valverde, L., Bouchon-Meunier, B., Yager, R.R. (eds.) IPMU 1992. LNCS, vol. 682, pp. 313–316. Springer, Heidelberg (1993)
26. Chen, G., Kerre, E.E., Vandenbulcke, J.: The axiomatic system of fuzzy functional dependency in a fuzzy relational data model. In: Shengkai, Z., Kaiqi, Z. (eds.) Information and Systems '92, Dalian Maritime University Publishing House, Dalian, China, pp. 876–880 (1992)

27. Chen, G., Vandenbulcke, J., Kerre, E.E.: On the lossless-join decomposition of relation scheme(s) in a fuzzy relational data model. In: Ayyub, B.M. (ed.) Proceedings ISUMA '93. Second International Symposium on Uncertainty Modeling and Analysis, pp. 440–446. IEEE Computer Society Press, Los Alamitos, U.S.A (1993)
28. Chen, G., Kerre, E.E., Vandenbulcke, J.: A computational algorithm for the FFD transitive closure and a complete axiomatization of fuzzy functional dependency. International Journal of Intelligent Systems 9, 421–439 (1994)
29. Chen, G., Kerre, E.E., Vandenbulcke, J.: Fuzzy normal forms and a dependency preserving decomposition into q-F3NF. In: Proceedings 3rd. IEEE International Conference on Fuzzy Systems, pp. 156–161. IEEE Computer Society Press, Los Alamitos (1994)
30. Chen, G., Kerre, E.E., Vandenbulcke, J.: The dependency – preserving decomposition and a testing algorithm in a fuzzy relational data model. Fuzzy Sets and Systems 72, 27–38 (1995)
31. Chen, G., Kerre, E.E.: An overview of fuzzy data models. In: Bosc, P., Kacprzyck, J. (eds.) Study in Fuzziness: Fuzziness in Database Management Systems, pp. 23–41. Physica-Verlag, Heidelberg (1995)
32. Chen, G., Vandenbulcke, J., Lemahieu, W., Kerre, E.E.: A scheme decomposition for fuzzy third normal forms to satisfy both dependency – preserving and lossless – join properties. In: Proceedings IFSA '95, Sao Paulo, Brazil, vol. 2, pp. 373–376.
33. Chen, G., Kerre, E.E.: Handling data redundancy and update anomalies in fuzzy relational databases. In: Ruan, D., D'hondt, P., Govaerts, P., Kerre, E. (eds.) Intelligent Systems and Soft Computing for Nuclear Science and Industry, pp. 233–242. World Scientific Publishing, Singapore (1996)
34. Chen, G., Kerre, E.E., Vandenbulcke, J.: Normalization based on fuzzy functional dependency in a fuzzy relational data model. Information Systems 21, 299–310 (1996)
35. Chen, G., Kerre, E.E., Vandenbulcke, J.: Introducing keys and integrity rules in a fuzzy relational data model. In: Proceedings 1996 IEEE International Conference on Systems, Man and Cybernetics, Beijing, China, October 14-17, 1996, vol. 1(4), pp. 332–335 (1996)
36. Chen, G., Kerre, E.E.: An extended Boyce-Codd normal form in fuzzy relational databases. In: Proceedings FUZZ-IEEE Conference, New Orleans, pp. 1546–1551 (1996)
37. Chen, G., Kerre, E.E.: Fuzzy functional dependencies and their roles in fuzzy databases. Proceedings 5th European Congress on Intelligent Techniques and Soft Computing 2, 1117–1121 (1997)
38. Chen, G., Wei, G., Kerre, E.E.: Fuzzy data mining: discovery of fuzzy generalized association rules. In: Bordogna, G., Pasi, G. (eds.) Recent Issues on Fuzzy Databases, pp. 45–66. Physica-Verlag, Heidelberg (2000)
39. Chen, G., Yan, P., Kerre, E.E.: Computationally Efficient Mining for Fuzzy Implication-Based Association Rules in Quantitative Databases. Int. J. General Systems 33(2-3), 163–182 (2004)
40. Groenemans, R., Kerre, E.E., De Cooman, G., Van Ranst, E.: Fuzzy database model based on quasi-order relations. Journal of Intelligent Information Systems 8, 227–243 (1997)
41. Groenemans, R., Van Ranst, E., Kerre, E.E.: Fuzzy relational calculus in land evaluation. GEODERMA 77, 283–298 (1997)
42. Cappelle, B., Kerre, E.E., Ruan, D., Vanmassenhove, F.: Characterization of binary operations on the unit interval satisfying the generalized modus ponens inference rule. Mathematica Pannonica 2(1), 105–121 (1991)

43. Kerre, E.E.: A comparative study of the behaviour of some popular fuzzy implication operators on the generalized modus ponens. In: Zadeh, L.A., Kacprzyck, J. (eds.) Fuzzy Logic for the Management of Uncertainty, pp. 281–296. John Wiley and Sons, New York (1992)
44. Ruan, D., Kerre, E.E.: On the extension of the compositional rule of inference. International Journal of Intelligent Systems 8, 807–817 (1993)
45. Ruan, D., Kerre, E.E.: Fuzzy implication operators and generalized fuzzy method of cases. Fuzzy Sets and Systems 54, 23–37 (1993)
46. Ruan, D., Kerre, E.E.: On the extended fuzzy chaining syllogism. In: Proceedings IFSA '95. Sixth International Fuzzy Systems Association World Congress, Sao Paulo, Brazil, vol. I, pp. 145–148 (1995)
47. Ruan, D., Kerre, E.E.: On if–then–else inference rules. In: Proceedings 1996 IEEE International Conference on Systems, Man and Cybernetics, Beijing, China, October 14-17, 1996, vol. 2(4), pp. 1420–1425 (1996)
48. De Cock, M., Bodenhofer, U., Kerre, E.E.: Modelling linguistic expressions using fuzzy relations. In: Proceedings 6th International Conference on Soft Computing. Iizuka 2000, Iizuka, Japan, October 1-4, 2000,CDROM, pp. 353–360 (2000)
49. De Cock, M., Nachtegael, M., Kerre, E.E.: Images under fuzzy relations: a masterkey to fuzzy applications. In: Ruan a.o., D. (ed.) Intelligent Techniques and Soft Computing in Nuclear Science and Engineering, pp. 47–54. World Scientific, Singapore (2000)
50. De Cock, M., Kerre, E.E.: The representation of linguistic hedges using fuzzy relational calculus. In: Smith, M.H., Gruver, W.A., Hall, L.O. (eds.) Proceedings Joint 9th IFSA World Congress and 20th NAFIPS International Conference, vol. 3, pp. 1608–1613 (2001)
51. De Cock, M., Kerre, E.E.: On (un)suitable fuzzy relations to model approximate equality. Fuzzy Sets and Systems 133, 137–153 (2003)
52. De Cock, M., Kerre, E.E.: Why fuzzy T-equivalence relations do not resolve the Poincaré paradox, and related issues. Fuzzy Sets and Systems 133, 175–180 (2003)
53. De Cock, M., Kerre, E.E.: Fuzzy modifiers based on fuzzy relations. Information Sciences 160, 173–199 (2004)
54. Radzikowska, A.-M., Kerre, E.E.: Algebraic characterisations of some fuzzy information relations. In: Atanassov, K., Hryniewicz, O., Kacprzyk, J. (eds.) Soft Computing Foundations and Theoretical Aspects, pp. 337–352 (2004)
55. Radzikowska, A.-M., Kerre, E.E.: A comparative study of fuzzy rough sets. Fuzzy Sets and Systems 126, 137–155 (2002)
56. Schockaert, S., Ahn, D., De Cock, M., Kerre, E.E.: Question answering with imperfect temporal information. In: Larsen, H.L., Pasi, G., Ortiz-Arroyo, D., Andreasen, T., Christiansen, H. (eds.) FQAS 2006. LNCS (LNAI), vol. 4027, pp. 647–658. Springer, Heidelberg (2006)
57. Schockaert, S., De Cock, M., Kerre, E.E.: Fuzzifying Allen's temporal interval relations. IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics 2007 (to appear)
58. De Baets, B., Kerre, E.E., Gupta, M.M.: Fundamentals of fuzzy mathematical morphology, part I: basic concepts. International Journal of General Systems 23, 155–171 (1995)
59. De Baets, B., Kerre, E.E., Gupta, M.M.: Fundamentals of fuzzy mathematical morphology, part II: idempotence, convexity and decomposition. International Journal of General Systems 23(4), 307–322 (1995)

60. Nachtegael, M., Kerre, E.E.: Decomposing and Constructing of Fuzzy Morphological Operations over alpha-cuts: continuous and discrete case. IEEE Transactions on Fuzzy Systems 8(5), 615–626 (2000)
61. Nachtegael, M., Kerre, E.E.: Connections between binary gray-scale and fuzzy mathematical morphologies. Fuzzy Sets and Systems 124, 73–86 (2001)
62. De Witte, V., Schulte, S., Nachtegael, M., Van der Weken, D., Kerre, E.E.: Vector morphological operators for colour images. In: Kamel, M., Campilho, A. (eds.) ICIAR 2005. LNCS, vol. 3656, pp. 667–675. Springer, Heidelberg (2005)
63. Wang, X., Cappelle, B., Ruan, D., Kerre, E.E.: Transitivity of fuzzy orderings based on pairwise comparisons. The Journal of Fuzzy Mathematics 3, 455–463 (1995)
64. Wang, X., Kerre, E.E.: Some relationships between different ordering procedures based on a fuzzy relation. In: Proceedings IPMU (7th), Paris, July 6-10, 1998, pp. 246–249 (1998)
65. De Baets, B., Fodor, J., Kerre, E.E.: Gödel representable fuzzy weak orders. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 7(2), 135–153 (1999)
66. Wang, X., Kerre, E.E.: Reasonable properties for the ordering of fuzzy quantities, Part I and II. Fuzzy Sets and Systems 118, 375–406 (2001)
67. Bodenhofer, U., De Cock, M., Kerre, E.E.: Openings and closures of fuzzy preorderings: theoretical basics and applications to fuzzy rule-based systems. Int. J. General Systems 32(4), 343–360 (2003)
68. Kerre, E.E.: Fuzzy approach to ECG diagnosis. In: Singh, M. (ed.) Encyclopedia of Systems and Control, vol. 2, pp. 1405–1407. Pergamon Press, Oxford (1987)
69. Kerre, E.E.: A walk through fuzzy relations and their application to information retrieval, medical diagnosis and expert systems. In: Ayyub, B., Gupta, M., Kanal, L. (eds.) Analysis and Management of Uncertainty, Theory and Applications, North–Holland, Amsterdam, pp. 141–151 (1992)
70. Kerre, E.E.: Fuzzy logic and medical diagnosis. In: Proceedings of the Symposium Clear Applications of Fuzzy Logic, Delft, pp. 160–174 (1991)
71. De Baets, B., Van de Walle, B., Kerre, E.E.: Transitivity relationships in fuzzy structures without incomparability. In: Chin, R., Naiman, A., Pong, T.-C., Ip, H.H.-S. (eds.) ICSC 1995. LNCS, vol. 1024, pp. 60–67. Springer, Heidelberg (1995)
72. De Baets, B., Van de Walle, B., Kerre, E.E.: Fuzzy preference structures and their characterization. The Journal of Fuzzy Mathematics 3, 373–381 (1995)
73. De Baets, B., Van de Walle, B., Kerre, E.E.: Fuzzy preference structures without incomparability. Fuzzy Sets and Systems 76, 333–348 (1995)
74. Van de Walle, B., De Baets, B., Kerre, E.E.: On the existence, construction and characterization of fuzzy preference structures. In: Proceedings IFSA '95. Sixth International Fuzzy Systems Association World Congress, Sao Paulo, Brazil, vol. II, pp. 141–144 (1995)
75. Van de Walle, B., De Baets, B., Kerre, E.E.: A plea for the use of Lukasiewicz triplets in the definition of fuzzy preference structures. Part I: General argumentation. Fuzzy Sets and Systems 97, 349–359 (1998)
76. De Baets, B., Van de Walle, B., Kerre, E.E.: A plea for the use of Lukasiewicz triplets in the definition of fuzzy preference structures. Part II: the identity case. Fuzzy Sets and Systems 99, 303–310 (1998)
77. Van de Walle, B., De Baets, B., Kerre, E.E.: Characterizable fuzzy preference structures. Annals of Operations Research 80, 105–136 (1998)

Golden Quadruplet: Optimization - Inequality - Identity - Operator

Seiichi Iwamoto

Faculty of Economics, Kyushu University
6-19-1 Hakozaki, Higashi, Fukuoka 812-8581, Japan
iwamoto@en.kyushu-u.ac.jp

Abstract. The Golden ratio is one of the most beloved numbers in human society. It is a symobol of combination of beauty and practical use. We consider the Golden ratio through four models —(i) optimization, (ii) inequality, (iii) identity and (iv) opertator —. We introduce the Golden matrices whose characteristic values are the Golden number and its conjugate. We show that the Golden matrices take an important role in the four models. Further the role is essentially equivalent.

1 Introduction

The Golden ratio is one of the most beautiful irrational numbers in science, technology and art, especially in mathematics, biology and architecture [2,3,7,8,9]. The human history proves it. We enjoy it in our daily life.

In this paper we show that the Golden number takes an important and beautiful part in mathematical models. We take four mathematical models — optimization, inequality, identity and operator —. These mutually related four are the basic mathematical tools. We show that if we place the Golden number in the center of the four models, a beautiful relation is observed. It is shown that optimization, inequality and identity are equivalent each other [4,5]. As the fourth equivalent model we consider operator, which is expressed by a 2×2 integer symmetric matrix.

In section 2, we introduce an elemental matrix which is called Golden. Section 3 shows a quadruplet associated with the Golden matrix. Section 4 shows the n-th power of the Golden matrix. In section 5, a quadruplet associated with the the n-th power matrix is shown. In the last section, we introduce another three Golden matrices and state their corresponding results.

2 Golden Matrix A

A real number

$$\phi = \frac{1+\sqrt{5}}{2} \approx 1.618$$

is called *Golden number*. It is the larger of the two solutions to quadratic equation (QE)

$$x^2 - x - 1 = 0. \tag{1}$$

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 14–23, 2007.

Sometimes QE (1) is called *Fibonacci* or *Golden*. The Golden QE has two real solutions: ϕ and its *conjugate* $\overline{\phi} := 1 - \phi$. We note that

$$\phi + \overline{\phi} = 1, \qquad \phi \cdot \overline{\phi} = -1.$$

Further we have

$$\phi^2 = 1 + \phi, \qquad \overline{\phi}^2 = 2 - \phi$$
$$\phi^2 + \overline{\phi}^2 = 3, \qquad \phi^2 \cdot \overline{\phi}^2 = 1.$$

Let us consider an integer symmetric matrix

$$A = \begin{pmatrix} 1 & -1 \\ -1 & 0 \end{pmatrix}$$

as an elemental matrix. The matrix A is called *Golden*. A reason will be clarified as follows. The characteristic equation of A is Golden:

$$\lambda^2 - \lambda - 1 = 0. \tag{2}$$

Eq.(2) has the two eigenvalues — the Golden number and its conjugate —

$$\lambda_1 = \phi, \qquad \lambda_2 = 1 - \phi = \overline{\phi} \tag{3}$$

with the corresponding eigenvectors with the Golden ratio

$$x_1 = c \begin{pmatrix} -\phi \\ 1 \end{pmatrix}, \qquad x_2 = c \begin{pmatrix} 1 \\ \phi \end{pmatrix} \tag{4}$$

where $-\infty < c < \infty$. For the normality, we take

$$c = \frac{1}{\sqrt{2 + \phi}}.$$

Then the eigenvectors $x_1,\ x_2$ are orthonormal.

3 The First Quadruplet

We consider an equivalence between four models — optimization, inequality, identity and operator — through the Golden matrix A.

Let us now take the quadratic form

$$(x, y) A \begin{pmatrix} x \\ y \end{pmatrix} = x^2 - 2xy$$

as an objective function. Our optimization problem is

$$\begin{array}{lll} & \text{Maximize and minimize} & x^2 - 2xy \\ \text{(OP)} & \text{subject to} & \text{(i)} \quad x^2 + y^2 = 1 \\ & & \text{(ii)} \quad -\infty < x, y < \infty. \end{array}$$

Theorem 1. (Optimization) *Then* (OP) *has the maximum value* $M = \phi$ *at the points* $(x^*, y^*) = \pm\dfrac{1}{\sqrt{2+\phi}}\,(\phi, -1\,)$ *and the minimum value* $m = \overline{\phi}$ *at the points* $(\hat{x}, \hat{y}) = \pm\dfrac{1}{\sqrt{2+\phi}}\,(1, \phi\,)$.

The following inequality and identity hold true.

Theorem 2. (Inequality) *It holds that*

$$\overline{\phi}(x^2+y^2) \le x^2 - 2xy \le \phi(x^2+y^2) \quad \text{on} \quad R^2.$$

The sign of right equality holds if and only if $x+\phi y = 0$. *The sign of left equality holds if and only if* $\phi x - y = 0$.

Theorem 3. (Identity) *It holds that*

$$x^2 - 2xy - \overline{\phi}(x+\phi y)^2 = \phi(x^2+y^2)$$
$$\overline{\phi}(x^2+y^2) - \overline{\phi}(\phi x - y)^2 = x^2 - 2xy.$$

The forementioned three statements are equivalent each other (see [1,5,6]).

Let the Golden matrix

$$A = \begin{pmatrix} 1 & -1 \\ -1 & 0 \end{pmatrix}$$

be associated with two auxiliary matrices

$$U = \phi I - A\,, \qquad V = A - \overline{\phi} I.$$

Both identity matrix and zero matrix are denoted by

$$I = \begin{pmatrix} 1\,0 \\ 0\,1 \end{pmatrix}, \qquad O = \begin{pmatrix} 0\,0 \\ 0\,0 \end{pmatrix}.$$

Then we have

$$U = \begin{pmatrix} -\overline{\phi}\ 1 \\ 1\ \ \phi \end{pmatrix}, \qquad V = \begin{pmatrix} \phi\ \ -1 \\ -1\ -\overline{\phi} \end{pmatrix}.$$

We note that

$$(x,y)U\begin{pmatrix} x \\ y \end{pmatrix} = -\overline{\phi}x^2 + 2xy + \phi y^2$$
$$= -\overline{\phi}(x+\phi y)^2 \ge 0$$

$$(x,y)V\begin{pmatrix} x \\ y \end{pmatrix} = \phi x^2 - 2xy - \overline{\phi}y^2$$
$$= -\overline{\phi}(\phi x - y)^2 \ge 0.$$

We write $P > Q$ (resp. $P \ge Q$) if and only if the subtracted matrix $P - Q$ is positive definite (resp. nonnegative definite). These are also written as $Q < P$ (resp. $Q \le P$). Then we have the following results.

Lemma 1.

$$\begin{cases} U \geq O \\ V \geq O. \end{cases}$$

Further we have

$$\begin{cases} -U + V = \begin{pmatrix} 1 & -2 \\ -2 & -1 \end{pmatrix} \\ U + V = (\phi - \overline{\phi})I. \end{cases}$$

Lemma 2.

$$\begin{cases} UV = O \\ U^2 = (\phi - \overline{\phi})U \\ V^2 = (\phi - \overline{\phi})V. \end{cases} \tag{5}$$

We say that U, V are *ϕ-quadratic* iff (5) is satisfied.

Theorem 4. (Operator) *It holds that*

$$\begin{cases} A = \phi I - U \\ A = \overline{\phi} I + V \end{cases} \quad \textit{where} \quad \begin{cases} U \geq O \\ V \geq O. \end{cases}$$

It is easily shown that Theorems 1, 2, 3 and 4 are equivalent. Thus we have a quadruplet of equivalent statements.

4 The n-th Power A^n

It is well known that the Golden number is closely related to the Fibonacci sequence. The Fibonacci sequence $\{a_n\}$ is defined by the difference equation

$$a_{n+2} - a_{n+1} - a_n = 0 \qquad a_0 = 0,\ a_1 = 1. \tag{6}$$

In fact, the Golden number and its conjugate are the limits of the ratio of adjacent terms in the Fibonacci sequence:

$$\begin{cases} \phi = \lim\limits_{n \to \infty} \dfrac{a_{n+1}}{a_n} \\ \overline{\phi} = \lim\limits_{n \to -\infty} \dfrac{a_{n+1}}{a_n}. \end{cases}$$

The Fibonacci sequence is tabulated in Table 1.

The number a_n is called *n-th Fibonacci number*. We note that

$$\begin{aligned} a_n > 0 \qquad & n = \cdots, -3, -1, 0, 1, 2, \cdots \\ a_n < 0 \qquad & n = \cdots, -6, -4, -2. \end{aligned} \tag{7}$$

Table 1. Fibonacci sequence $\{a_n\}$

n	-13	-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1
a_n	233	-144	89	-55	34	-21	13	-8	5	-3	2	-1	1	0	1

n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
a_n	0	1	1	2	3	5	8	13	21	34	55	89	144	233	377	610	987	1597

We remark that the linearization

$$\begin{cases} \phi^n &= a_{n-1} + a_n\phi \\ \overline{\phi}^n &= a_{n-1} + a_n\overline{\phi} \end{cases}$$

holds true. A matrix form is the following.

Theorem 5. (Linearization)

$$A^n = a_nA + a_{n-1}I$$

where a_n is the n-th Fibonacci number.

Thus we have

$$A^n = \begin{pmatrix} a_{n+1} & -a_n \\ -a_n & a_{n-1} \end{pmatrix}.$$

5 The n-th Quadruplet

Now let us consider a quadruplet associated with the n-th power matrix A^n.

Theorem 6. (n-Operator)

$$\begin{cases} A^n = \phi^nI - a_nU \\ A^n = \overline{\phi}^nI + a_nV \end{cases} \quad \textit{where} \quad \begin{cases} U = \phi I - A \geq O \\ V = A - \overline{\phi}I \geq O. \end{cases}$$

where a_n is the n-th Fibonacci number.

Proof. The proof is by induction or through linearization. □

Corollary 1.

$$\overline{\phi}^nI \leq A^n \leq \phi^nI \qquad n = \cdots, -3, -1, 0, 1, 2, \cdots$$

$$\overline{\phi}^nI \geq A^n \geq \phi^nI \qquad n = \cdots, -6, -4, -2.$$

Corollary 2.

$$\begin{cases} A^n = \dfrac{1}{2}\left(\phi^n + \overline{\phi}^n\right)I + \dfrac{a_n}{2}\begin{pmatrix} 1 & -2 \\ -2 & -1 \end{pmatrix} \\ \left(\phi^n - \overline{\phi}^n\right) = a_n(\phi - \overline{\phi}). \end{cases}$$

Let us now take an optimization problem associated with matrix

$$A^n = \begin{pmatrix} a_{n+1} & -a_n \\ -a_n & a_{n-1} \end{pmatrix}.$$

We consider the optimization problem

$$\begin{array}{ll} & \text{Maximize and minimize} \quad a_{n+1}x^2 - 2a_nxy + a_{n-1}y^2 \\ (\mathrm{OP_n}) & \text{subject to} \quad \text{(i)} \quad x^2 + y^2 = 1 \\ & \qquad\qquad\quad \text{(ii)} \quad -\infty < x, y < \infty. \end{array}$$

From Therem 6 together with (7), we have three equivalent statements as follows.

Theorem 7. (n-Optimization) (i) *Let an integer n be negative odd or nonnegative. Then* $(\mathrm{OP_n})$ *has the maximum value $M = \phi^n$ at the points $(x^*, y^*) = \pm\frac{1}{\sqrt{2+\phi}}(\phi, -1)$ and the minimum value $m = \overline{\phi}^n$ at the points $(\hat{x}, \hat{y}) = \pm\frac{1}{\sqrt{2+\phi}}(1, \phi)$.*

(ii) *Let an integer n be negative even. Then* $(\mathrm{OP_n})$ *has the minimum value $m = \phi^n$ at the points $(\hat{x}, \hat{y}) = \pm\frac{1}{\sqrt{2+\phi}}(\phi, -1)$ and the maximum value $M = \overline{\phi}^n$ at the points $(x^*, y^*) = \pm\frac{1}{\sqrt{2+\phi}}(1, \phi)$.*

Theorem 8. (n-Inequality) (i) *Let an integer n be negative odd or nonnegative. Then it holds that*

$$\overline{\phi}^n(x^2 + y^2) \le a_{n+1}x^2 - 2a_nxy + a_{n-1}y^2 \le \phi^n(x^2 + y^2) \quad \text{on} \quad R^2.$$

(ii) *Let an integer n be negative even. Then it holds that*

$$\overline{\phi}^n(x^2 + y^2) \ge a_{n+1}x^2 - 2a_nxy + a_{n-1}y^2 \ge \phi^n(x^2 + y^2) \quad \text{on} \quad R^2.$$

In either case, the sign of right equality holds if and only if $x + \phi y = 0$. The sign of left equality holds if and only if $\phi x - y = 0$.

Theorem 9. (n-Identity) *It holds that*

$$a_{n+1}x^2 - 2a_nxy + a_{n-1}y^2 - a_n\overline{\phi}(x + \phi y)^2 = \phi^n(x^2 + y^2)$$

$$\overline{\phi}^n(x^2 + y^2) - a_n\overline{\phi}(\phi x - y)^2 = a_{n+1}x^2 - 2a_nxy + a_{n-1}y^2.$$

We note that

$$\begin{pmatrix} a_{n+1} & -a_n \\ -a_n & a_{n-1} \end{pmatrix} = a_n \begin{pmatrix} 1 & -1 \\ -1 & 0 \end{pmatrix} + a_{n-1} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

This yields the identity.

$$a_{n+1}x^2 - 2a_nxy + a_{n-1}y^2 = a_n(x^2 - 2xy) + a_{n-1}(x^2 + y^2).$$

Therefore, an optimization of quadratic form

$$a_{n+1}x^2 - 2a_nxy + a_{n-1}y^2$$

under a condition $x^2 + y^2 = c$ is reduced to optimization of

$$x^2 - 2xy$$

under the same condition.

Now we may take the *first* optimization problem

$$\begin{array}{lll} & \text{Maximize and minimize} & x^2 - 2xy \\ (\mathrm{OP}_1) & \text{subject to} \quad \text{(i)} & x^2 + y^2 = 1 \\ & \qquad\qquad\quad \text{(ii)} & -\infty < x, y < \infty. \end{array}$$

Then (OP_1) has the maximum value $M = \phi$ at the points $(x^*, y^*) = \pm\dfrac{1}{\sqrt{2+\phi}} \times (\phi, -1)$ and the minimum value $m = \overline{\phi}$ at the points $(\hat{x}, \hat{y}) = \pm\dfrac{1}{\sqrt{2+\phi}}(1, \phi)$.

Let an integer n be negative odd or nonnegative. Then the n-th optimization problem $(\mathrm{OP_n})$ has the maximum value $M = a_n\phi + a_{n-1}$ at the same points (x^*, y^*) and the minimum value $m = a_n\overline{\phi} + a_{n-1} = -a_n\phi + a_{n+1}$ at the same points $(\hat{x}, \hat{y})$. Let n be negative even. Then $(\mathrm{OP_n})$ has the minimum value $m = a_n\phi + a_{n-1}$ at the points (x^*, y^*) and the maximum value $M = a_n\overline{\phi} + a_{n-1} = -a_n\phi + a_{n+1}$ at the points $(\hat{x}, \hat{y})$.

6 Golden Matrices B, C and F

We have considerd some properties of the elemental matrix

$$A = \begin{pmatrix} 1 & -1 \\ -1 & 0 \end{pmatrix}.$$

Now we specify the similar results for another three elemental matrices.

$$B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}, \quad C = \begin{pmatrix} 0 & -1 \\ -1 & 1 \end{pmatrix}, \quad F = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}.$$

Let us now introduce the associated three matrices as follows.

$$J = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad K = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad L = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

Lemma 3. *The first two are symmetric and the third is skew-symmetric:*

$$J' = J, \quad K' = K, \quad L' = -L.$$

The square are

$$J^2 = I, \quad K^2 = I, \quad L^2 = -I.$$

Thus the inverse matrices are

$$J^{-1} = J, \quad K^{-1} = K, \quad L^{-1} = -L.$$

The three matrices B, C and F are connected to the matrix A through J, K and L, respectively.

Lemma 4.

$$B = JAJ, \quad C{=}KAK, \quad F = L'AL,$$

$$A = JBJ, \quad A{=}KCK, \quad A = L'FL.$$

6.1 Matrix B

The elemental matrix $B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$ has the following properties.

Corollary 3. *It holds that*

$$\begin{cases} B^n = \phi^n I - a_n U \\ B^n = \overline{\phi}^n I + a_n V \end{cases}$$

where two nonnegative definite matrices

$$U = \begin{pmatrix} -\overline{\phi} & -1 \\ -1 & \phi \end{pmatrix}, \qquad V = \begin{pmatrix} \phi & 1 \\ 1 & -\overline{\phi} \end{pmatrix}$$

are ϕ-quadratic. Therefore,

$$\overline{\phi}^n I \le B^n \le \phi^n I \qquad n = \cdots, -3, -1, 0, 1, 2, \cdots$$

$$\overline{\phi}^n I \ge B^n \ge \phi^n I \qquad n = \cdots, -6, -4, -2.$$

Thus, for matrix B, we have a corresponding Theorem 6 (n-Operator), Theorem 7 (n-Optimization), Theorem 8 (n-Inequality) and Theorem 9 (n-Identitity). A golden quadruplet holds true for matrix B.

6.2 Matrix C

The elemental matrix $C = \begin{pmatrix} 0 & -1 \\ -1 & 1 \end{pmatrix}$ has the following properties.

Corollary 4. *It holds that*

$$\begin{cases} C^n &= \phi^n I - a_n U \\ C^n &= \overline{\phi}^n I + a_n V \end{cases}$$

where two nonnegative definite matrices

$$U = \begin{pmatrix} \phi & 1 \\ 1 & -\overline{\phi} \end{pmatrix}, \qquad V = \begin{pmatrix} -\overline{\phi} & -1 \\ -1 & \phi \end{pmatrix}$$

are ϕ-quadratic. Therefore,

$$\begin{aligned} \overline{\phi}^n I \;&\le\; C^n \;\le\; \phi^n I \qquad n = \cdots, -3, -1, 0, 1, 2, \cdots \\ \overline{\phi}^n I \;&\ge\; C^n \;\ge\; \phi^n I \qquad n = \cdots, -6, -4, -2. \end{aligned}$$

Thus, for matrix C, we have a correponding golden quadruplet: optimization – inequality – identitity – operator.

6.3 Matrix *F*

The elemental matrix $F = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$ has the following properties.

Corollary 5. *It holds that*

$$\begin{cases} F^n &= \phi^n I - a_n U \\ F^n &= \overline{\phi}^n I + a_n V \end{cases}$$

where two nonnegative definite matrices

$$U = \begin{pmatrix} \phi & -1 \\ -1 & -\overline{\phi} \end{pmatrix}, \qquad V = \begin{pmatrix} -\overline{\phi} & 1 \\ 1 & \phi \end{pmatrix}$$

are ϕ-quadratic. Therefore,

$$\begin{aligned} \overline{\phi}^n I \;&\le\; F^n \;\le\; \phi^n I \qquad n = \cdots, -3, -1, 0, 1, 2, \cdots \\ \overline{\phi}^n I \;&\ge\; F^n \;\ge\; \phi^n I \qquad n = \cdots, -6, -4, -2. \end{aligned}$$

Thus, a golden quadruplet holds true for matrix F.

References

1. Bellman, R.E.: Introduction to Matrix Analysis. McGraw-Hill, New York (1970)
2. Beutelspacher, A., Petri, B.: Der Goldene Schnitt 2., überarbeitete und erweiterte Auflange. Elsevier GmbH, Spectrum Akademischer Verlag, Heidelberg (1996)

3. Dunlap, R.A.: The Golden Ratio and Fibonacci Numbers. World Scientific Publishing Co. Pte. Ltd (1977)
4. Iwamoto, S.: Cross dual on the Golden optimum solutions. In: Proceedings of the Workshop in Mathematical Economics, RIMS Kokyu Roku No. 1443, pp. 27–43, Kyoto University (2005)
5. Iwamoto, S.: The Golden trinity — optimility, inequality, identity —. In: Proceedings of the Workshop in Mathematical Economics, RIMS Kokyu Roku No. 1488, pp. 1–14, Kyoto University (2006)
6. Iwamoto, S.: The Golden optimum solution in quadratic programming. In: Proc. of the Fourth Intl Conference on Nonlinear Analysis and Convex Analysis (NACA 2005), Okinawa, June-July 2005, pp. 109–205. Yokohama Publishers, Yokohama (2007)
7. Iwamoto, S., Yasuda, M.: Dynamic programming creates the Golden ratio, too. In: Proceedings of the Workshop Decision Models and Mathematical Models under Uncertainty, RIMS Kokyu Roku No. 1477, pp. 136–140, Kyoto University (2006)
8. Iwamoto, S., Yasuda, M.: Golden optimal path in discrete-time dynamic optimization processes. In: The 15-th International Conference on Difference Equations and Applications (ICDEA 2006), Kyoto, July, 2006 (to appear)
9. Walser, H.H.: Der Goldene Schnitt. B.G. Teubner, Leipzig (1996)

Algorithms for String Pattern Discovery

Hideo Bannai

Department of Informatics, Kyushu University,
774 Moto-oka, Nishiku, Fukuoka 819–0395, Japan
bannai@i.kyushu-u.ac.jp

Abstract. Pattern discovery from string data is an important problem with many applications. In this paper, we give a brief overview of our work on the *optimal correlated pattern discovery problem*, which integrates numerical attribute information into the string pattern discovery process.

1 Introduction

Due to the phenomenal growth of the Internet, computer storage and computational power, we are now able to store and utilize massive amounts of data. Although the types of data are diverse, a large portion of the data can be considered as *strings*, for example, web pages and biological sequences such as DNA, RNA, and proteins. Discovering meaningful *string patterns* from these data is an important topic of study, with numerous applications [1,2].

Unlike conventional data attributes which are explicitly given, attributes on string data are implicit, and various *pattern classes* can be considered. Good patterns which characterize a set of strings can be very useful, since it is not very difficult to interpret the knowledge encapsulated in them. Patterns can also be used as a basis for classifying and characterizing new and unknown strings, by determining whether or not the pattern matches the string. Our research group has been actively studying this area, starting from the development of the BONSAI system [3], which was one of the early systems for the classification of string data. We have then explored the subproblem of finding the optimal string pattern that distinguishes between positive and negative string sets, for various pattern classes [4,5,6,7,8,9].

Due to recent technical advances such as the development of microarrays [10], vast amounts of numerical measurements that are related to genomic sequences have been produced. For example, it is possible to measure the expression levels of all the genes of a given organism. To output more relevant and reliable results, methods that effectively integrate different sources of information into the data analysis process have come to attract attention [11,12]. To this end, we have considered a new formulation of pattern discovery called *correlated pattern discovery*. We consider the case where we are given a single set of sequences, and each sequence is assigned a numeric attribute value. The problem is to find the pattern whose occurrence in the sequences are correlated with the numeric attribute value [13,14,15,16].

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 24–29, 2007.

In this paper, we briefly review our basic framework for optimal correlated pattern discovery.

2 The Optimal Pattern Discovery Problem

Let Σ be a finite alphabet. An element of Σ^* is called a *string*. Strings x, y, and z are said to be a *prefix*, *substring*, and *suffix* of string $w = xyz$, respectively. The length of a string w is denoted by $|w|$. The empty string is denoted by ε, that is, $|\varepsilon| = 0$. Let $\mathbf{R}$ represent the set of real numbers.

A *pattern class* is a pair $\mathcal{C} = (\Pi, \psi)$, where Π is a set called the *pattern set* and $\psi : \Pi \times \Sigma^* \to \{\texttt{true}, \texttt{false}\}$ is a *pattern matching function*. An element $p \in \Pi$ is called a *pattern*. For a pattern p and string $t \in \Sigma^*$, we say that *the pattern p of class $\mathcal{C}$ matches text t* if $\psi(p, t) = \texttt{true}$, and *$p$ of class $\mathcal{C}$ does not match t*, otherwise. For example, the *substring pattern class* is defined as $\mathcal{C}_{\text{substr}} = (\Sigma^*, \psi_{\text{substr}})$, where $\psi_{\text{substr}}(p, t) = \texttt{true}$ *iff* p is a substring of t.

For a given pattern class $\mathcal{C} = (\Pi, \psi)$ and a set of strings $S = \{s_1, \ldots, s_m\}$, let $M_\psi(p, S)$ denote the set of indices of strings in S that p matches. That is,

$$M_\psi(p, S) = \{i \mid \psi(p, s_i) = \texttt{true}, i = 1, \ldots, m\}.$$

The optimal classificatory pattern discovery problem is defined as follows:

Problem 1 (Optimal classificatory pattern discovery). For a given pattern class $\mathcal{C} = (\Pi, \psi)$, string sets $S, T \subseteq \Sigma^*$ $(S \cap T = \emptyset)$, and function *score*, find $\hat{p} \in \Pi$ such that:

$$\hat{p} = \arg\max_{p \in \Pi} score(|M_\psi(p, S)|, |S|, |M_\psi(p, T)|, |T|).$$

We require that the scoring function is dependent only on the classificatory strength of a pattern, represented by the number of strings in each set, as well as the number of strings in each set that the pattern matches. Examples of such scoring functions are: Gini index, information entropy gain, and chi-squared statistic.

Now suppose that for a given set of strings $S = \{s_1, \ldots, s_m\}$, we are given a set of numeric attribute values $R = \{r_1, \ldots, r_m\}$, where each $r_i \in \mathbf{R}$ represents some aspect of string $s_i \in S$. Without loss of generality, assume that $r_1 \leq \ldots \leq r_m$.

Problem 2 (Optimal correlated pattern discovery). For a given pattern class $\mathcal{C} = (\Pi, \psi)$, a set of strings $S = \{s_1, \ldots, s_m\}$, a set of numeric attributes $R = \{r_1, \ldots, r_m\}$, and function *score*, find the pattern $\hat{p} \in \Pi$ such that:

$$\hat{p} = \arg\max_{p \in \Pi} score(|M_\psi(p, S)|, |S|, \sum_{i \in M_\psi(p,S)} r_i, \sum_{i \in \{1,\ldots,m\}} r_i).$$

Here, we require that the scoring function is dependent on the number of strings and the total sum of the numeric attribute values, as well as the number of strings that the pattern matches, and the total sum of the numeric attribute

values for strings that the pattern matches. Examples of such scoring functions are: interclass variance and Wilcoxon rank sum test statistic (provided that r_i is converted to rank information).

It is not difficult to see that the optimal classificatory pattern discovery problem is a special case of the optimal correlated pattern discovery problem. For a given instance of the optimal classificatory pattern discovery problem, let $S' = S \cup T = \{s_1, \ldots, s_m\}$, and let $R = \{r_1, \ldots, r_m\}$ where $r_i = 1$ if $s_i \in S$ and $r_i = -1$ if $s_i \in T$. Noticing that $|M_\psi(p, S')| = |M_\psi(p, S)| + |M_\psi(p, T)|$, $\sum_{i \in M_\psi(p,S')} r_i = |M_\psi(p, S)| - |M_\psi(p, T)|$, and $\sum_{i \in \{1,\ldots,m\}} r_i = |S| - |T|$, we have that

$$|M_\psi(p, S)| = \frac{|M_\psi(p, S')| + \sum_{i \in M_\psi(p,S')} r_i}{2},$$
$$|M_\psi(p, T)| = \frac{|M_\psi(p, S')| - \sum_{i \in M_\psi(p,S')} r_i}{2},$$
$$|S| = \frac{|S'| + \sum_{i \in \{1,\ldots,m\}} r_i}{2},$$
$$|T| = \frac{|S'| - \sum_{i \in \{1,\ldots,m\}} r_i}{2}.$$

Therefore, the optimal correlated pattern for string set S, attribute set R, and scoring function $score'(w, x, y, z) = score\left(\frac{w+y}{2}, \frac{w-y}{2}, \frac{x+z}{2}, \frac{x-z}{2}\right)$ is equivalent to the optimal classificatory pattern for string set S, T, and scoring function $score$.

In what follows, we will denote $score(w, y) = score(w, x, y, z)$ since x and z are constant for a given instance of the problem.

3 Algorithm Overview

Finding the optimal correlated pattern basically amounts to enumerating all possible patterns in the pattern class as candidates, and selecting the pattern that gives the best score.

For the substring pattern class, the problem can be solved very efficiently making use of the suffix tree data structure [17]. A suffix tree for a given string is a compacted trie of all the suffixes of the string, and can be represented and constructed in linear time and space. Although the number of substrings of a given string is quadratic in its length, the number of candidate patterns can be restricted to a linear number. Furthermore, it is possible to calculate the score of each candidate pattern in constant time using algorithmic techniques [18,14].

Unfortunately, the problem has been shown or are believed to be NP-hard for most other useful pattern classes [19,20]. In the worst case, we must enumerate an exponential number of candidate patterns. We have therefore developed branch and bound heuristics so that the problem can be solved in a practical amount of time. The following property holds for reasonable scoring functions:

Property 1. For any $0 \leq x \leq m$, there exists $y \in \mathbf{R}$ such that:

$$score(x, y') \leq score(x, y'') \text{ for any } y \leq y' < y'' \text{ and}$$
$$score(x, y') \geq score(x, y'') \text{ for any } y' < y'' \leq y$$

For such scoring functions, we can use the following lemma to prune the search space of candidate patterns. Notice that for most reasonable pattern classes, any pattern $q \in \Pi$ obtained by elongating pattern $p \in \Pi$ will have $M_\psi(p, S) \supseteq M_\psi(q, S)$.

Lemma 1 (Pruning Lemma). *Consider a given pattern class $\mathcal{C} = (\Pi, \psi)$, a set of strings $S = \{s_1, \ldots, s_m\}$, a set of numeric attributes $R = \{r_1, \ldots, r_m\}$, and function score. Consider a fixed pattern $p \in \Pi$, and let $M_\psi(p, S) = \{i_{j_1}, \ldots, i_{j_k}\}$. Then, for any $q \in \Pi$ such that $M_\psi(p, S) \supseteq M_\psi(q, S)$,*

$$score(|M_\psi(q, S)|, \sum_{i \in M_\psi(q,S)} r_i)$$
$$\leq \max\{\{score(l, \sum_{h=1}^{l} r_{i_{j_h}}) \mid l = 0, \ldots, k\} \cup \{score(l, \sum_{h=1}^{l} r_{i_{j_{k-h+1}}}) \mid l = 0, \ldots, k\}\}$$

By using this lemma, we have shown in several computational experiments that the problem can be solved in a reasonable amount of time for several pattern classes [13,15,16].

4 Concluding Remarks

We have briefly presented the basic framework of our optimal correlated pattern discovery algorithms. We have recently extended the framework further to consider more general matching functions $\psi : \Pi \times \Sigma^* \to \mathbf{R}$. For example, $\psi(p, t)$ could represent the number of times a given pattern $p \in \Pi$ occurs in the string t [21].

Acknowledgments

This paper represents joint works with many of my colleagues, especially Setsuo Arikawa, Satoru Miyano, Masayuki Takeda, Ayumi Shinohara, Kenta Nakai, Shunsuke Inenaga, Kohei Hatano, and Heikki Hyyrö.

References

1. Brazma, A., Jonassen, I., Eidhammer, I., Gilbert, D.: Approaches to the automatic discovery of patterns in biosequences. J. Comput. Biol. 5, 279–305 (1998)
2. Shinohara, A.: String pattern discovery. In: Ben-David, S., Case, J., Maruoka, A. (eds.) ALT 2004. LNCS (LNAI), vol. 3244, pp. 1–13. Springer, Heidelberg (2004)

3. Shimozono, S., Shinohara, A., Shinohara, T., Miyano, S., Kuhara, S., Arikawa, S.: Knowledge acquisition from amino acid sequences by machine learning system BONSAI. Transactions of Information Processing Society of Japan 35(10), 2009–2018 (1994)
4. Hirao, M., Inenaga, S., Shinohara, A., Takeda, M., Arikawa, S.: A practical algorithm to find the best episode patterns. In: Jantke, K.P., Shinohara, A. (eds.) DS 2001. LNCS (LNAI), vol. 2226, pp. 435–440. Springer, Heidelberg (2001)
5. Hirao, M., Hoshino, H., Shinohara, A., Takeda, M., Arikawa, S.: A practical algorithm to find the best subsequence patterns. Theoretical Computer Science 292(2), 465–479 (2002)
6. Shinohara, A., Takeda, M., Arikawa, S., Hirao, M., Hoshino, H., Inenaga, S.: Finding best patterns practically. In: Arikawa, S., Shinohara, A. (eds.) Progress in Discovery Science. LNCS (LNAI), vol. 2281, pp. 307–317. Springer, Heidelberg (2002)
7. Takeda, M., Inenaga, S., Bannai, H., Shinohara, A., Arikawa, S.: Discovering most classificatory patterns for very expressive pattern classes. In: Grieser, G., Tanaka, Y., Yamamoto, A. (eds.) DS 2003. LNCS (LNAI), vol. 2843, pp. 486–493. Springer, Heidelberg (2003)
8. Inenaga, S., Bannai, H., Shinohara, A., Takeda, M., Arikawa, S.: Discovering best variable-length-don't-care patterns. In: Lange, S., Satoh, K., Smith, C.H. (eds.) DS 2002. LNCS, vol. 2534, pp. 86–97. Springer, Heidelberg (2002)
9. Inenaga, S., Bannai, H., Hyyrö, H., Shinohara, A., Takeda, M., Nakai, K., Miyano, S.: Finding optimal pairs of cooperative and competing patterns with bounded distance. In: Suzuki, E., Arikawa, S. (eds.) DS 2004. LNCS (LNAI), vol. 3245, pp. 32–46. Springer, Heidelberg (2004)
10. Shalon, D., Smith, S., Brown, P.: A DNA microarray system for analyzing complex DNA samples using two-color fluorescent probe hybridization. Genome Res. 6(7), 639–645 (1996)
11. Bussemaker, H.J., Li, H., Siggia, E.D.: Regulatory element detection using correlation with expression. Nature Genetics 27, 167–171 (2001)
12. Conlon, E.M., Liu, X.S., Lieb, J.D., Liu, J.S.: Integrating regulatory motif discovery and genome-wide expression analysis. Proc. Natl. Acad. Sci. 100(6), 3339–3344 (2003)
13. Bannai, H., Inenaga, S., Shinohara, A., Takeda, M., Miyano, S.: A string pattern regression algorithm and its application to pattern discovery in long introns. Genome Informatics 13, 3–11 (2002)
14. Bannai, H., Inenaga, S., Shinohara, A., Takeda, M., Miyano, S.: Efficiently finding regulatory elements using correlation with gene expression. Journal of Bioinformatics and Computational Biology 2(2), 273–288 (2004)
15. Bannai, H., Hyyrö, H., Shinohara, A., Takeda, M., Nakai, K., Miyano, S.: Finding optimal pairs of patterns. In: Jonassen, I., Kim, J. (eds.) WABI 2004. LNCS (LNBI), vol. 3240, pp. 450–462. Springer, Heidelberg (2004)
16. Bannai, H., Hyyrö, H., Shinohara, A., Takeda, M., Nakai, K., Miyano, S.: An $O(N^2)$ algorithm for discovering optimal Boolean pattern pairs. IEEE/ACM Transactions on Computational Biology and Bioinformatics 1(4), 159–170 (2004) (special issue for selected papers of WABI2004)
17. Weiner, P.: Linear pattern matching algorithms. In: Proc. 14th IEEE Annual Symp. on Switching and Automata Theory, pp. 1–11 (1973)

18. Hui, L.: Color set size problem with applications to string matching. In: Apostolico, A., Galil, Z., Manber, U., Crochemore, M. (eds.) Combinatorial Pattern Matching. LNCS, vol. 644, pp. 230–243. Springer, Heidelberg (1992)
19. Miyano, S., Shinohara, A., Shinohara, T.: Which classes of elementary formal systems are polynomial-time learnable? In: Watanabe, O., Yokomori, T. (eds.) ALT 1999. LNCS (LNAI), vol. 1720, pp. 139–150. Springer, Heidelberg (1999)
20. Jiang, T., Li, M.: On the complexity of learning strings and sequences. Theoretical Computer Science 119(2), 363–371 (1993)
21. Bannai, H., Hatano, K., Inenaga, S., Takeda, M.: Practical algorithms for pattern based linear regression. In: Hoffmann, A., Motoda, H., Scheffer, T. (eds.) DS 2005. LNCS (LNAI), vol. 3735, pp. 44–56. Springer, Heidelberg (2005)

Voting in the Medieval Papacy and Religious Orders

Ian McLean[1], Haidee Lorrey[2], and Josep Colomer[3]

[1] Nuffield College, Oxford
[2] Berklee College of Music
[3] Higher Council of Scientific Research (CSIC) and Pompeu Fabra University, Barcelona
iain.mclean@nuffield.ox.ac.uk, haideeusa@yahoo.com,
josep.colomer@gmail.com

Abstract. We take institutions seriously as both a rational response to dilemmas in which agents found themselves and a frame to which later rational agents adapted their behaviour in turn. Medieval corporate bodies knew that they needed choice procedures. Although the social choice advances of ancient Greece and Rome were not rediscovered until the high middle ages, the rational design of choice institutions predated their rediscovery and took some new paths. Both Ramon Llull (ca 1232-1316) and Nicolaus of Cusa (a.k.a Cusanus; 1401-64) made contributions which had been believed to be centuries more recent. Llull promotes the method of pairwise comparison, and proposes the Copeland rule to select a winner. Cusanus proposes the Borda rule, which should properly be renamed the Cusanus rule.

Voting might be needed in any institution ruled by more than one person, where decisions could not simply be handed down from above. Medieval theologians no doubt believed that God's word was handed down from above; but they well knew that they often had to decide among rival human interpretations of it. The Church faced its own decision problem every time a new Pope needed to be elected. Bodies not directly in the hierarchy of the Church had to evolve their own decision procedures. The chief such bodies were commercial and urban corporations; religious orders; and universities.

The disagreement between Llull and Cusanus raises the issue: should voting be regarded as a method of aggregating judgments or as a method of aggregating interests? In the former interpretation (only), voting procedures are a solution to a problem of approximate reasoning. There is an unknown, true state of affairs (for medieval thinkers, divine will). A voting procedure aggregates unreliable individual perceptions of the will of God to a more reliable group judgment of it. In the rougher world of Cusanus, and probably of electors to the papacy and to Dogeships, only at most lip service is paid to the will of God, and voting is a process of aggregating interests.

Keywords: Voting, Medieval Papacy, Religious Orders.

1 Introduction

We take institutions seriously as both a rational response to dilemmas in which agents found themselves and a frame to which later rational agents adapted their behaviour in

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 30–44, 2007.

turn. Modalities of thought differ; rationality does not. We aim to show that the institutions we discuss represent rational responses to problems. Medieval corporate agents knew that they needed choice procedures. Although the social choice advances of ancient Greece and Rome were not rediscovered until the high middle ages, the rational design of choice institutions predated their rediscovery and took some new paths.

Although we normally think of voting as a mechanism for expressing tastes or preferences, there is an alternative framework in which it represents *judgments.* This was the framework not only for medieval social choice but also for much of classical social choice up to and including Condorcet. Ramon Llull saw voting precisely as a mechanism for approximate reasoning. Humans had an imperfect understanding of God's will. They differed in their understanding of the correct course of action, where the correct course of action was to carry out God's will. Voting was a procedure for aggregating these imperfect individual judgments into a more reliable group judgment. His follower Nicolaus Cusanus proposed a different choice procedure, for rival electors from different parts of Europe, with different territorial interests, to choose an Emperor. In Llull's interpretation (only), voting procedures are a solution to a problem of approximate reasoning. There is an unknown, true state of affairs (for Llull, the will of God). In the rougher world of Cusanus, and probably of electors to the papacy and to Dogeships, only at most lip service is paid to the will of God, and voting is a process of aggregating interests.

The strategy of this paper is as follows. Section 2 introduces and interprets the texts of Llull and Cusanus. Sections 3 and 4 review medieval practice in elections of popes and officials of monastic orders. Section 5 concludes, reviewing the history of the two rival conceptions of voting in medieval Europe.

2 Theory: Llull and Cusanus

The classical Greek contribution to the theory and practice of social choice lies in the development of juries and other random choice procedures of judgment aggregation. In democratic Athens in the era of Pericles, the governing institutions were Assembly, Council, and juries. The Assembly was the meeting of all citizens. The executive was the Council, whose membership was chosen by lot and rotation, so that any citizen might be president of Athens for a day. Juries were bodies of (typically) 501, 1001, or 1501 members, numbers being odd to avoid ties. Voting played a relatively minor role, the most common recorded case being votes on proposals to ostracise (banish) citizens.

Voting was a more developed institution in republican Rome. Pliny the Younger discusses parliamentary procedure in the Roman Senate (Farquharson 1969 *passim*; Riker 1986 chapter 7; McLean and Urken 1995, chapter 2). He is clearly writing in a context of commonly understood rules. However, he is thinking more in an *interests* than in a *judgments* framework, since the context is what we now label strategic, or sophisticated, voting. Pliny tried to justify his novel stratagem (of replacing binary voting by ternary, in the vain hope of gaining a strategic advantage) in terms of appeals to ancient parliamentary authority.

The theorists we consider here made an entirely fresh start. After Pliny, social choice theory did not appear again in the Western world until the sudden emergence of sophisticated themes in the work of Ramon Llull (ca. 1232--1316). Llull, and probably Nicolaus of Cusa (Cusanus; Niklaus von Kues, 1401-64), did not know the Greek and Latin writings on social choice. But Llull worked at the frontier of Christian and Islamic scholarship. Thus the issue may have been discussed during the golden age of Arabic scholarship between the seventh and thirteenth centuries C.E.; but we have not found any evidence for that. However, progress beyond Pliny requires algebra and some knowledge of combinations and permutations, and these were Arabic inventions. Medieval Europe saw little discussion of democracy. The framework within which people either thought or had to pretend that they thought was one in which a choice either conformed to the will of God or it did not: a binary choice. Llull and Cusanus saw that the case of multiple candidates threw up new problems.

Llull was a native of Palma (Mallorca), which in his time was part of a Catalan economic empire with links across the western Mediterranean, including mainland Spain, recently under Islamic rule, and north Africa, still under Islamic rule. Llull wrote copiously in Catalan, Arabic, and Latin. At the age of about thirty he became a devout Christian and devoted the rest of his life to missionary work and theology. He wrote poetry and a novel (the first in any Western European language), and copious writings in mathematics and logic. In his autobiography, written in 1311, he wrote that when he was converted he had a vision that God had called him to write "a book, the best book in the world, against the errors of unbelievers" (Llull 1311, in Bonner 1985, I:15). This was to be the *Ars Generalis* with which Llull struggled for the rest of his life, in between journeys to North Africa to convert the Moors and equally unsuccessful visits to successive popes urging them to set up language schools for missionaries. He frequently introduced mathematical arguments for the truth of Christianity into his theological works. His theory of voting appears three known places: his novel *Blanquerna*, written in Catalan between 1282 and 1287; a newly discovered paper *Artifitium electionis personarum* (discovered by Perez Martinez 1959, but first discussed by Hägele and Pukelsheim 2001; date unknown but before 1283) and a short paper entitled *De arte eleccionis*, written in 1299. We describe first the more picturesque text, then the more scholarly ones.

Blanquerna is the beloved son of Evast and Aloma. When he reaches the age of eighteen, he decides to become a hermit despite his mother's anguished pleas to stay with his parents. His mother sends Natana to him in the hope of persuading him to stay; instead, he persuades Natana to renounce her possessions as well and enter a nunnery. In due course she becomes its abbess; meanwhile, Blanquerna becomes successively a monk, an abbot, a bishop (reluctantly), and the pope, before renouncing everything again and becoming a hermit. The story gives Llull the opportunity to introduce homely dialogs illustrating the deadly sins and the virtues of the Christian life. In one of his anecdotes, the Abbess has died, and the nuns are deciding what to do.

> All the sisters wanted to elect their abbess by their usual electoral method, but Natana said that she had heard of a new electoral method, which consisted in art [in other words, Llull's General Art] and figures. (McLean and Urken 1995, p. 71).

Natana told the twenty sisters that they should first elect seven electors, each to nominate (presumably seven) names excluding herself. The seven electors should compare the candidates with each other according to four conditions, namely, which of them best loves and knows God, which of them best loves and knows the virtues, which of them knows and hates most strongly the vices, and which is the most suitable person.

She goes on to describe the case of nine candidates (viz., the seven electors and two outsiders). The electors should compare the candidates two by two, and for each pair determine which they judge to be the more God-loving, virtuous, vice-hating, and suitable:

> Therefore, taking this number as an example, 36 cells *[cambres]* will be produced in which the votes of each candidate will appear. The candidate to be elected should be the one with the most votes in the most cells. (adapted from McLean and Urken 1995, pp. 71-3).

The theory of elections recurs in later chapters of *Blanquerna*, as when Blanquerna is elected abbot "according to the manner of election whereby Natana had been elected abbess." Later still, he is proposed for a bishopric. He does not want it because it would mean giving up the contemplative life. Most of the electors nevertheless vote for him on the advice of the retiring bishop, but his enemy the archdeacon "opposed the holding of an election according to the art". One takes place "without the art," but it leads to a dispute, the majority electing Blanquerna and the minority the archdeacon. Both sides go to Rome, where the pope rules in favor of the reluctant Blanquerna. Thus people who oppose the correct art of elections come to a suitably sticky end (Peers 1926, chaps. 24, 60, and 67).

Hägele and Pukelsheim (2001) date *Artifitium electionis personarum* to between 1274 and 1283. It is in a Vatican MS which they translate into English for the first time. Like the later *De Arte Eleccionis,* it opens with a matrix for pairwise comparisons of candidates, in this case with 16 candidates and hence 120 pairs. The crux of the procedural description is in these sentences, describing how each pair is to be scored:

> Et omnes responderint et eligerint prout eis uidebitur[. F]iat vnus punctus in littera attribuata illi person[a]e qu[a]e plures uoces habuerit. Qui punctus fiat ipsi littere in qualibet figurarum existentium in locis diuersis. Si uero vna habuerit tot uoces ut altera fiat in qualibet littera ipsius camere punctus vnus et hoc in qualibet figurarum.
>
> *And all shall respond, and shall elect as it appears [fit] to them. [Then] a point is placed by the letter assigned to the person who has the most votes. Such a point is marked in each of the figures existing at the distinct locations. If now one [person] has as many votes as another, then a point is placed by both letters of this cell, and this [is done] in each of the figures* [in the vote matrix]. (Transcription and translation by Hägele and Pukelsheim 2001 with minor edits by IM and HL)

The electoral procedure in *De Arte Eleccionis* was devised, Llull tells us, at Paris on 1 July, 1299. In his autobiography he complained that nobody understood him when he lectured in Paris because of his "Arabic way of speaking" (Bonner 1985,

vol. 1, pp. 29, 38).) *De Arte Eleccionis* contains a proposed method of election, similar to that in *Artifitium electionis personarum,* except that it does not use all the pairwise comparisons in the vote matrix.

The electoral method in *Blanquerna* is a two-stage procedure. Like Condorcet and the US Federalists five centuries later, Llull seems to wish to compromise between democracy and giving a more decisive voice to better qualified electors. The election is to be made on multiple (four) criteria. Llull may have realized that multiple-criterion decision making can lead to difficulties in aggregating from individual to social orderings. And it is a method of exhaustive pairwise comparisons. Votes are to be placed in 36 *cambres* (cells). These represent the 36 combinations of two candidates from nine---as it would now be written $n(n - 1)/2$ for $n = 9$. Llull has Natana state that "the candidate to be elected should be the one with the most votes in the most cells". How is this phrase to be interpreted? There are two natural interpretations, one of which makes *Blanquerna* an anticipation of the Borda rule and the other an anticipation of the Copeland rule. (Riker 1982 p. 79; McLean and Urken 1995, p. 18; Klamler 2005).

On the first interpretation, the phrase "in the most cells" is redundant, since each candidate will have votes in just eight cells. These votes are simply summed, and the candidate with the highest aggregate is elected. This is, as is now well known, exactly a Borda count in which zero points are awarded for a last place, one for a second-to-last, and so on up to n-1 for a top place. Borda pointed out this equivalence in his paper of 1770 (in McLean and Urken 1995, p. 87). On this interpretation, the following passage about ties refers to ties in the Borda count:

> One of the sisters asked her, "If it turns out that some candidates have as many votes as each other in the cells, what procedure does the art recommend?" Natana replied, "The art recommends that these two or three or more should be judged according to art alone. It should be found out which of these best meets the four aforementioned conditions, for she will be the one who is worthy to be elected".

The Copeland rule has regard to the number of majorities each candidate has, not to their size, individually or in aggregate. It selects the candidate who wins the largest number of contests. If there is no cycle, the Copeland winner is the same as the Condorcet winner. If there is a top cycle, there is no Condorcet winner and a set of Copeland winners numbering three or more. Is this what Llull meant? On this interpretation, the whole phrase, "The candidate to be elected should be the one with the most votes in the most cells" is then an exact instruction to select the Copeland winner, reading the first "most" as "more"; and the passage about ties is an instruction on how to select a unique winner from the Copeland set if that contains more than one member. But note that the Copeland set cannot contain just two members unless there are ties on individual pairs arising from abstention, individual indifference, or an even number of voters, none of which Llull seems to allow in this text since he seems to insist on an odd number of voters, each with a strong ordering.

There is no such ambiguity in either *Artifitium electionis personarum* (AEP) or in *De Arte Eleccionis*. (DAE) The passage translated above from AEP seems to clinch matters in favor of Copeland. The winner of each pair is counted and a mark put against the winner's name. The passage translated makes it clear that this is a group,

not an individual, mark. That individual is then given one point (*unus* punctus) for each majority win. The winner must be the candidate with the largest number of *puncti*, that is, the Copeland winner. If there is a tie, both candidates are awarded one point, so that there could be a Copelandoid set of size 2. The procedure Llull recommends in the two papers, unlike that described in *Blanquerna*, is a Condorcet pairwise comparison procedure. It uses matrix notation, previously thought to have been first used by C. L. Dodgson (Lewis Carroll) nearly six centuries later. Because the winning candidate must have beaten at least one other, it cannot select a Condorcet loser, and if a Condorcet winner exists, it will select him or her. The *AEP* procedure could, but the *DAE* procedure could not, detect the existence of cycles because in *DAE* not every comparison in the matrix is actually used in selecting the winner. The fact that Llull makes this (to us) retrograde step in the later procedure suggests that he did not discover cycling.

Thus Llull does not deserve the scornful treatment he gets in modern histories of mathematics and logic. Combinatorics, which he was probably one of the first mathematicians in the West to import from the Arab world, fascinated him endlessly and fueled the magnificent but impossible dream of the General Art. Llull believed that applying successive pairwise combinations of virtues could lead one to "demonstrating the truth of the holy Catholic faith through the use of necessary reasons to those who are ignorant of it" (Bonner 1985, 69). This led Donald Michie to label Llull "one of the most inspired madmen who ever lived" (Gardner 1982, ix). Martin Gardner has written that Llull's life was "much more fascinating than his eccentric logic. . . . Llull's mistake . . . was to suppose that his combinatorial method had useful applications to subject matters where today we see clearly that it does not apply" (Gardner 1982, xiv, 18). However, the application to voting rules is an entirely appropriate application of the mathematics of combinations, not repeated until 1785. *Providing* that voting is regarded as a procedure for aggregating imperfect individual judgments into a more reliable group judgment, exhaustive pairwise comparison is an appropriate method.

Nicolaus Cusanus read *De arte eleccionis* and may have been the transcriber of the only known copy: Cusanus was born in 1401 beside the river Moselle. He studied first at Heidelberg, then at Padua, where he gained his doctorate in 1423, then at Cologne. Padua was one of the leading intellectual centers of Europe, and Llull's mathematical and theological works were on the curriculum there (Sigmund 1963, 22-35). Llull had a dangerous reputation: anybody whose ideas were as hard to follow as his risked being suspected of heresy. But the intellectual climate in northern Italy was more open than elsewhere. Cusanus was active in the conciliar movement of his time. The Council of Constance (1414-1417) addressed the Great Schism in the papacy that had lasted since 1378; it succeeded in ousting all three of the current contenders for the title of pope and electing one of its choice. It featured a weighted voting scheme (voting by nations) to ensure that the Italian electors did not carry the day by sheer force of numbers. Most council members were bishops, and Italy had the largest number of bishoprics.

Cusanus's *De concordantia catholica* was written while he was attending the Council of Basel, which opened in 1431; Cusanus was an active member from 1432 to 1434. *De concordantia catholica* defends the rights of councils to elect popes, and it discusses voting procedures for electing a Holy Roman Emperor in chapters

36 to 38 of Book III. Cusanus first discusses the need to prevent *practicas absurdissimas et inhonestissimas* (the most absurd and dishonest practices) and notes that because particular electors come from particular districts, *turpiter foedatae electiones per iniustas pactiones fieri dicuntur* (elections are said to be disgracefully rigged by means of unjust pacts) (McLean and Urken 1995, pp. 77-8).

Cusanus' scheme is just the Borda count, giving 1 for a last place and so on up to *n* for a top place. Although Cusanus knew of the existence of *Blanquerna*, he had probably not read it. A mention of it exists in a handlist of Llull's work in Cusanus's library (Honecker 1937b, 570--1), but the work itself does not, and the library is believed to have survived complete. The only known copy of *De Arte Eleccionis* comes from this library, transcribed in what is believed to be Cusanus's handwriting. Thus Cusanus knew Llull's Condorcet scheme of public voting; but he proposed instead a Borda scheme with secret voting.

We believe that Cusanus' rejection of Llull is deliberate. We refer to the phrase *quoniam omnes comparationes omnium personarum et omnes mixturae et syllogismi per unumquemque ex electoribus factibiles in hoc modo includuntur*. "For this method takes account of all comparisons of candidate to candidate---in whatever groupings or combinations---that any elector can make," in Cusanus's paragraph 540. The use of *syllogismi* in this sense is highly unusual. A syllogism involves at least three elements (major premise, minor premise and conclusion). In using the word, Cusanus may have had in mind that a voter's transitive ordering of three elements (I prefer *A* to *B* and *B* to *C*; therefore I prefer *A* to *C*) is fully captured by the Borda count but not always by the scheme of *De Arte Eleccionis*. The theory of voting involves pairwise comparisons, as Llull had seen. But it does not only involve pairwise comparisons.

Both writers wish to eliminate strategic voting, but they make opposite recommendations. Llull, in *Blanquerna*, writes about the members of a religious order voting selecting their own leader. The electors are all known to each other and must continue to live together after the vote. A voter will then be constrained by her fellow voters' knowledge of her preferences. In general, this is the argument for open voting in committees where the members must trust one another if business is to be done. It is an argument traditionally accepted in the direct democracy, with open voting, of some Swiss cantons (Barber 1985). Cusanus writes about a body of electors meeting once only and suspicious of one another's strategic voting intentions before the election starts. Increasing the amount of information about others' votes available to each voter increases the opportunities and incentives for strategic voting of a logrolling kind. It was presumably in part to prevent this that the Council of Constance had voted by nations. Llull sees voting as an aggregation of judgments; Cusanus, writing from bitter experience, sees it as an aggregation of interests.

Cusanus goes on to show that his method may also be applied to votes on propositions when more than two possibilities exist and contrasts it with the simple binary procedure in use in Venice for yes or no propositions and for elections (Sigmund 1991, 307, 580). The Venetian procedure for electing a doge, in use between 1268 and 1797, has been regarded as pointlessly complicated. Lines (1986), however, shows that "a great deal of the tedious complication . . . served only to ensure the impossibility of forecasting just who would be in the Quarantuno [the 41 electors]" (Lines 1986, 156). The actual election stage was approval voting with a lower bound. The electors voted on each of the ten candidates separately and could

vote in favor of or against as many as they chose; the winner would be the candidate with the most favorable votes, as long as he got more than 25.

Thus we find four of the main solution concepts of modern voting theory---the Borda rule, the Condorcet principle, the Copeland rule, and approval voting---in use in medieval Europe. We should not be surprised to find intelligent discussion of voting schemes first appearing in the West in the Middle Ages. The experience of the Great Schism in the papacy showed that elections could not be restricted to two candidates. All orders of monks, nuns and friars had to make their own rules for electing their superiors; since they were entirely separate from the ordinary parish clergy, there was no hierarchy except their own to choose their leaders. This is the situation addressed by Llull.

3 Practice: The Papacy

Unanimity was first conceived as the only rule that could reveal God's will, but it led to frequent deadlocks, conflicts and schisms. It was replaced with the qualified majority rule of two-thirds in 1179. However the particular mixture of devices that were adopted by the Church during the late Middle Ages did not tend to produce quick and consensual decisions, but frequently led to uninformed coalition-building with surprising results.

In early centuries, the Pope was elected by a multicameral college. First, the lay members of the Roman Church proposed candidates. Second, the clergy proceeded to reduce or to enlarge that list. Finally, the sixteen bishops of the Roman province met and decided. This sequence was reflected in the motto of pope Leo I (440-61) *vota civium, testimonia populorum, honoratum arbitrium, electio clericorum*. This frequently produced conflicts and schisms. Before the Emperor had officially accepted Christianity, there was at least one simultaneous election of two different popes (in 250, after 18 months of deadlock). Afterwards, elections of pairs of popes by different factions of the Church provoked intervention of Roman troops in 366 and 418. These conflicts put the Church under political protection. The emperor Honorius ruled in 420 that if two popes were elected, neither would be valid and a new election would be called in which divine judgment, as revealed by unanimity (*divinum judicium et universitatis consensus*), would be required.

The primacy of the Pope over political powers was doctrinally asserted in pope Gregory VII's bull *Dictatus Papae* (1075). But enforcing it required intensive legislative activity, starting with Gratian (1139-40), followed by four enlarged canon law codes in 100 years.

The Church required an orderly succession of its monarch. New rules emerged from successive decisions on partial aspects of the question. The first was a papal bull of 1059 which excluded laymen from the election of the Pope (Nicolaus II, *In nomine Domini*). The role of the Emperor was once again reduced to mere acceptance of the Church's decision. Although secular rulers tried to continue exercising their veto-right against certain candidates for pope, this now had to be implemented by way of some faction of cardinals. Royal or imperial 'anti-popes' ceased to exist after 1122. This did not avert conflict. Three 12th-century elections produced a total of eight antipopes to

only nine 'official' popes in less than fifty years. Even without direct imperial appointments, these schisms emerged because the voters could not reach unanimous agreement.

Several procedures to manufacture unanimity were implemented, known as 'acclamation', 'scrutiny', and 'compromissum'. Elections by 'acclamation' were rare and, even according to some participants, enthusiastic and threatening roaring of crowds induced them, not initial coincidence of voters around a single candidate. (For the election of Gregory VII in 1073, so reported by the pope himself, see Robinson, 1990: 59-60).

'Compromissum' consisted in delegating the decision to a small commission when unanimous agreement could not be reached. However, delegation should be adopted by unanimity of those entitled to vote, specifying the rule to be followed by the delegates, and this was not a frequent resource either. The most common procedure was 'scrutiny', that is, voting, but new intellectual devices had to be implemented to create apparent unanimity where it did not exist. The most discussed of these was the *sanior et maior pars*, the 'sounder and greater part'. *Sanior* referred to the priority given to cardinal-bishops, to candidates' merit and to voters' merits, zeal or dignity (including age or seniority in post, and hierarchy). All these qualities were considered factors for the choice of 'the best' candidate. But the 'sanior pars' often did not coincide with the 'maior pars'. In elections of bishops or abbots such disputes were usually submitted to some arbitrator, such as the metropolitan bishop or even the Pope, but no such arbiter existed for papal elections (See authorities cited in Colomer and McLean 1998, p. 2).

Such conflicts led to the adoption of two-thirds majority rule by Pope Alexander III in 1179. The rule of two-thirds had previously been used in the election of some abbots. Two-thirds and other qualified-majority rules were also used in several Italian communes of the Middle Ages. Alexander III spent six months in Venice in 1177, forging reconciliation with the Emperor Frederick Barbarossa who had supported the 'anti-pope'.

Alexander decreed.

> ***Concerning the election of the supreme pontiff***
>
> …We decree, therefore, that if, by chance, some hostile man sowing discord among the cardinals, full concord cannot be attained with regard to constituting a pope; and, with the two thirds which agree, the other third be unwilling to agree, or presume of itself to ordain someone else: he shall be considered Roman pontiff who shall be elected and received by two thirds…..
>
> Moreover if anyone is elected to the office of pope by fewer than two thirds--unless greater concord is attained, he shall by no means be accepted, and shall be subject to the aforesaid penalty if he is unwilling to humbly abstain. From this, however, let no prejudice to the canonical and other ecclesiastical decrees arise, with regard to which the opinion of the greater and the sounder part [*maior et sanior pars*] should prevail; for when a doubt arises with regard to them, it can be defined by the judgement of a higher power. But in the Roman church, special decrees are made because recourse cannot be had to a higher power. (Doeberl, iv. p 253)

It seems clear that the basic aim of the qualified majority of two-thirds was to induce the formation of a sufficiently large coalition of cardinals. A two-thirds winner would tend to require a previous negotiation between supporters of different candidates, probably around compromise solutions. As some contemporary analyses noted, once a candidate was elected, the losers would need to persuade a majority of the winner's original supporters to change their mind. Faced with this requirement, it was reasonable to expect that the losing coalition would not fight on (see also Saari, 1994: 15-16).

Caplin and Nalebuff (1988) have shown that the rule of 64% guarantees a single winner under conditions of 'concavity' in voter preferences. This means that, when more voters prefer intermediate candidates than the average of those favoring extremes, there exists an unbeatable proposal, and furthermore no cycles are possible. In general, the majority rule needed to avoid cycles and ensure existence of an unbeatable proposal in a n-dimensional issue space is no higher than $1- [n/(n+1)]^n$. This ratio is 55% for two dimensional spaces, 57% for three-dimensional spaces, and, being increasing in n, its limit is $1 - (1/e)$, which is just under 64%. In a simple aggregation of interests, there can be up to many dimensions as candidates. This means that the 2/3 rule used to elect the Pope produces a stable outcome if Caplin and Nalebuff's conditions hold.

After adopting the rule of two-thirds, popes and canonists tended to agree that the 'maior pars' is always, by definition, also the 'sanior pars'. As pope Pius II summarized (on his own election in 1458): "What is done by two thirds of the sacred college, that is surely of the Holy Ghost, which may not be resisted" (in Gragg and Gabel, 1959: 88). The 2/3 rule is, for the first time, explicitly defended as a judgmental aggregation rule of approximate reasoning - even if it was reached by a completely different route.

The qualified majority requirement produced the desired stability effects but it had predictable consequences. The electors in 1216, 1241, 1243, 1261, 1265 and 1268-70 took several months to reach a decision, having to resort to commissions in several cases. In two of these elections, (1216 and 1243), the civil authorities reacted to cardinals' slowness by locking them up. In 1241 the head of civil administration in Rome locked them up in an old unhygienic building, guarded by police, but he only elicited a decision by threatening to have the corpse of the dead pope exhumed and shown publicly in full papal regalia after two years of the vacancy. In 1270, when two years had passed without an agreement, the public besieged the cardinals in the episcopal palace, removed the roof of the palace and allowed nothing but bread and water to be sent in. A new pope was elected on this occasion by compromissum after a record vacancy of 34 months (Vauchez, 1990: 522-3). Thus the two-thirds rule produced efficacious and rather stable outcomes, at the price of long delays in decision-making. This is now recognized as a classic trade-off in social choice.

The experiences of locking cardinals up led pope Gregory X to adopt a new procedure for their seclusion, known as the Conclave (Latin: 'with-key'), which was approved by the council of Lyon in 1274 (*Ubi periculum*). It aimed to obtain a quick decision, and to prevent strategic maneuvering in the election of the Pope. Similar institutions had been established in the Dominican constitution of 1228, as well as in communes such as Venice and Piacenza, respectively in 1229 and 1233 (Ruffini Avondo, 1925; Ullmann, 1972).

The cardinals gathered together, each with no more than one servant, in a closed papal palace, whose doors were walled up and watched by soldiers; they were to lead a life in common in a single room; to have no communication with the outer world; food was to be supplied to them through a guarded window; the menu was restricted from the fourth day on and reduced to bread, water and wine after the ninth day; the cardinals received no income until they reached a collective decision. Although some of these provisions were later softened, they created strong and increasing incentives for the cardinals to reach a common decision. Many cardinals fell ill and several died in conclave, precipitating agreement among the remaining participants. All side-payments, coercion or explicit pacts between cardinals were forbidden under penalty of excommunication and annulment of the election; they must keep silence during the election and afterwards. These rules made exchanges and formation of large coalitions very difficult and often promoted agreed outcomes on the basis of the immediate, apparent appeal of some candidate rather than on careful evaluation of his merits or religious fervor.

The first papal election under this procedure, in 1276, was made in one single day. The following popes suspended the application of this procedure, whereupon long delays reappeared: more than seven months in 1277, six months in 1281, almost eleven months in 1288, and 27 months in 1292-94. This evidence of conclaves' efficacy moved the pope elected in 1294, Celestine V, to re-establish it. Successful conclaves of one or a few days have become normal since then, including in 2006. For the rules of conclave see the authorities cited by Colomer and McLean 1998, p. 14.

4 Practice: Monastic Orders

Monastic orders faced the same problem as the papacy. They were not directly subject to papal control at all times, and were required to choose their own leaders and secure an unbroken succession whenever a leader died. The papal bull *Exiit qui seminat*, promulgated by Pope Nicholas III in 1279, contains provisions for the Franciscan order that illustrates just how reliant on self-determination and tradition the succession issue could be. As the bull notes, outlining the accepted procedure for choosing a new master of the order,

> Besides the friars of the aforesaid order doubting in regard to that which is said in the rule, that with the decease of the minister general there is to be an election of a successor by the ministers provincial and guardians[1] (*custodes*) in the Pentecost chapter, whether it is fitting that the multitude of all the *custodes* come together to the general chapter, or whether, so that everything be managed with greater tranquillity, it may be able to suffice that some from each province, who would vote in the name of others, would take part, We give this answer that namely the *custodes* of each province are to appoint one from [among] themselves, whom they are to send with their minister provincial on their own behalf to the chapter, committing their votes and

[1] This could mean 'guardian' in any of several senses, e.g., one appointed to watch over good order generally, or more specifically to supervise votes and elections.

> powers to the same, because, when they have appointed [him] by themselves, even We reckon a statute of this kind to have been approved, because also [Our] predecessor, Gregory IX, in a case of this kind is said to have responded in this manner. (*Exiit qui seminat,* Nicholas III. Our translation from the Latin text transcribed from the Registers of Nicholas III, p. 232-241, #564)

Many of the later monastic rules exhibit more sophisticated choice procedures.

The rule of St Benedict (6th century) was the first to be codified, and all subsequent orders until the Dominicans (early 13th century) followed the Benedictine rule with minor tweaks. All new orders had to deal with the succession crisis following the death of their founder. The rule of St. Benedict makes it clear that officials that were simply appointed, rather than elected by a majority, might cause dissent within an abbey--or the entire order.

> It happens all too often that the constituting of a Prior gives rise to grave scandals in monasteries. For there are some who become inflated with the evil spirit of pride and consider themselves second Abbots. By usurping power they foster scandals and cause dissensions in the community. Especially does this happen in those places where the Prior is constituted by the same Bishop or the same Abbots who constitute the Abbot himself. What an absurd procedure this is can easily be seen; for it gives the Prior an occasion for becoming proud from the very time of his constitution, by putting the thought into his mind that he is freed from the authority of his Abbot. (Rule of St. Benedict. St. Benedict's rule for monasteries, tr. Leonard Doyle OSB, Collegeville, MN, 2001. Chapter 65: On the Prior of the Monastery, Apr. 22 - Aug. 22 - Dec. 22)

Another example is that of the Gilbertines, founded by Gilbert of Sempringham some time before 1147, when he travelled to the Cistercian headquarters at Citeaux in order to derive a constitution from theirs. As the Gilbertines were a double order, of both monks and nuns, he had to devise a more complex choice procedure. It was assumed in typical Benedictine procedure that decisions would be unanimous, but in the case of a difference, a majority of 3 to 1 sufficed, any huge differences were referred to the magister (head of the order). (The Gilbertine constitution is in University of Oxford, Bodleian Library, MS Douce 136).

The most elaborate constitution was that of the Dominicans, first written in 1216, and revised in 1228 (Galbraith 1925, especially 5, 33, 46, 64, 103, 114, 226---36). They give a much greater role than earlier constitutions to internal democracy. In the early years of the Order, friars acquired the suffrage immediately upon profession. Later, friars had to wait first one, then two, then ultimately 4 years after profession to receive voting privileges.

Later in the 13th century, this concept of democracy gained the authority of Aquinas, through his rediscovery and commendation of Aristotle.

> Question 105. The Various Forms of Government, and their Fusion in the Government Given by God to the Jews. (Art. 1)
>
> With respect to the right ordering of power in a city or nation, two points must be considered: the first is that all should in some respect participate in the government…. So the best ordering of power within a city or a kingdom is

> obtained when there is one virtuous head who commands over all; and who has under him others who govern virtuously; and when, furthermore, all participate in such government, both because all are eligible, and because all participate in the election of those who rule…. For Moses and his successors governed their people as sole heads over all: but they elected to their assistance seventy two Elders according to virtue: as it is said (Deuteronomy I:15): 'And I took out of your tribes men, wise and honourable, and appointed them rulers'. And this was aristocracy. But it was democratic in the sense that they were elected from the whole people, for it is said in Exodus (XVIII:21): 'Seek out from the whole people wise men,' etc.; and also in the sense that the people elected them, for it is said in Deuteronomy (I:13): 'Let me have from among you wise and understanding men,' etc. So it is clear that there was an excellent ordering of authority in the Law [of the Old Testament]. (Aquinas 1948, pp 148-151)

This is what we are accustomed to think of as a Schumpeterian conception of democracy, which should perhaps be relabelled Thomist. Democracy entails the right to elect the sovereign. But while in place, the sovereign's authority is absolute. Clearly, such a doctrine justified the papal constitution, and that of any order that gave its magister sovereign authority. Typically, however, the Dominicans were ahead of the pack. Their constitution foresaw the problem of an incompetent magister. The Master General of the Order held office for life, but the general chapter had the right to impeach him. There were four acceptable reasons:

1 Crime
2 Causing disunity or harm to the Order
3 Inept administration
4. Inability to effectively perform his duties (illness, senility, etc).

Before impeachment, the general chapter was supposed to take one final step, which was to ask for his resignation. If he refused to resign, the impeachment process to formally depose him would take place.

Other self-governing bodies in medieval Europe included communes (such as Venice) and universities (such as Bologna, Paris, Oxford, and Cambridge - including their constituent colleges). They too needed election rules such that they could elect their own heads without outside interference. Their constitutions await detailed study. Just like the papal, imperial, Gilbertine, or Dominican electors, the writers of college constitutions needed a device to prevent a succession crisis. The senior fellow is always an identifiable person (*quae in ecclesia vacante prius fuit recepta*, in Llull's words). The old formula *maior et sanior pars* could very easily be amended, or corrupted, to *maior et senior pars*. Seniority can be held to breed wisdom. This confusion is as old as Llull and as new as the Statutes of Nuffield College, Oxford (1958), which entrust the procedure to elect a new Warden to the senior - not to the wisest - of the fellows.

5 Conclusion

The two modern conceptions of voting both occur in medieval choice theory, and medieval practice. Conception #1 is both epistemic and stochastic. Conception #2 is

neither. In first conception, the task of voting is to aggregate imperfect human judgments to the most reliable achievable group judgment of an unknowable true state of affairs. For medieval choice theorists, that unknown state of affairs was the will of God. For Condorcet, reviving choice theory in the 18th century, it was the truth of an allegation that a jury was called upon to decide – hence Condorcet's formulation has come to be known as his 'jury theorem'. This conception of voting is still unfamiliar to most voters, but an example where it applies is the task of determining, given that one or more sensors have indicated a fault in a piece of equipment, whether it is likelier that the equipment is faulty or that the sensor(s) have shown a false negative.

But reality kept on intruding. Electors to the papacy abandoned the unanimity rule in favour of the 2/3 rule, but then found that to avoid deadlock they had to impose privation, open voting, and rules against trade-offs and side-payments. The formula 'maior et sanior pars' was unstable. As early as St Benedict, rule-makers discovered that every defeated minority clamed to be 'sanior' than the majority. Intriguers like Llull's fictional archdeacon obviously abounded. Nobody seriously pretended, in Cusanus' day, that election of a Holy Roman Emperor was a process of finding the will of God. Rather, it was a process of finding a candidate who was acceptable to a sufficient number of electors. The Borda (Cusanus) rule chooses the candidate who on average ranks highest on electors' schedules. Unfortunately, it is highly manipulable, which is probably why Cusanus recommends voting by secret ballot.

Black's (1958) rediscovery of Condorcet (1785) reintroduced the idea that there are two rival conceptions of voting – as the aggregation of judgments and as the aggregation of interests. In an ordinary election, as Black (1958, p. 163) says, 'the phrase "the probability of the correctness of a voter's opinion" seems to be without definite meaning. In a jury vote it does have meaning.

The two leading solution concepts for elections and juries faced with $n > 2$ options are the Condorcet rule (with its extension, the Copeland rule), and the Borda rule. The Condorcet rule is 'Choose the option that wins each of its pairwise contests against each of the others'. The Copeland extension is 'Choose the option that scores the largest number of pairwise victories'. The Borda rule is 'Choose the option with the highest average rank'. These solution concepts may be applied to either conception of the nature of elections. For instance, Condorcet found to his embarrassment that where individual voter reliabilities are low, the Borda rule is more likely than the Condorcet rule to find the most probable solution to a jury problem (Black 1958, p. 170). We have shown that they are centuries older than the thinkers whose names they bear. Llull proposed the Copeland rule and may have proposed the Borda rule. Cusanus proposed the Borda rule. It may be too late to rename them, but it is not too late to hail these medieval discoveries.

References

Aquinas, T.: Selected Political Writings. In: D'Entreves, A.P. (ed.) Translated by J.G. Dawson Oxford, Blackwell (1948)

Barber, B.: Strong Democracy. Basic Books, New York (1985)

Black, D.: The Theory of Committeees and Elections. Cambridge University Press, Cambridge (1958)

Bonner, A. (ed.): Selected works of Ramon Llull, vol. 2. Princeton University Press, Princeton, NJ (1985)
Caplin, A., Nalebuff, B.: On 64%-Majority Rule. Econometrica 56(4), 787–814 (1988)
Colomer, J., McLean, I.: Electing popes: approval balloting and qualified-majority rule. Journal of Interdisciplinary History 29(1), 1–22 (1998)
Condorcet, M.J.A.N.C., de marquis.: Essai sur l'application de l'analyse á la probabilité des décisions rendues á la probabilité des voix. Imprimerie Royale, Paris (1785)
Farquharson, R.: Theory of Voting. Blackwell, Oxford (1969)
Galbraith, G.R.: The constitution of the Dominican order, pp. 1216–1360. Manchester University Press, Manchester (1925)
Gardner, M.: Logic machines and diagrams, 2nd edn. Harvester, Brighton (1982)
Gragg, F.A., Gabel, L.C. (eds.): Memoirs of a Renaissance Pope. The Commentaries of Pius II. Putnam's Sons, New York (1959)
Hägele, G., Pukelsheim, F.: Llull's writings on electoral systems. Studia Lulliana 41, 3–38 (2001)
Honecker, M.: Llullus-Handschriften aus dem Besitz des Kardinals Nikolaus von Cues: nebst einer Beschreibung der Llullus-Texte in Trier und einem Anhang über den wiederaufgefundenen Traktat De Arte Electionis. In: Gesammelte Aufsätze zur Kulturgeschichte Spaniens, edited by Finke,H.P.I., vol. § 6, pp. 252–309. Verlag der Aschendorffschen Verlagsbuchhandlung, Münster in Westfalen (1937a)
Honecker, M.: Ramon Lulls Wahlvorschlag Grundlage des Kaiserwahlplanes bei Nikolaus von Cues? Historisches Jahrbuch 57, 563–574 (1937b)
Kallen, G. (ed.): Nicolai de Cusa opera omnia, vol. 14. Felix Meiner, Hamburg (1964)
Klamler, C.: The Copeland rule and Condorcet's principle. Economic Theory 25(3), 745–749 (2005)
Lines, M.: Approval voting and strategy analysis: A Venetian example. Theory and Decision 20, 155–172 (1986)
McLean, I., Urken, A.B.: Classics of Social Choice. The University of Michigan Press, Ann Arbor (1995)
Nuffield College, Oxford, Statutes: Oxford University Press for the College, Oxford(1958)
Peers, E.A. (ed.): Blanquerna: A 13th century romance translated from the Catalan of Raymond Lull. Jarrolds, London (1926)
Perez Martinez, L.: El Ars notandi y el Ars electionis, dos obras desconocidas de Ramon Llull. Estudios Lulianos 3, 275–278 (1959)
Riker, W.H.: Liberalism against Populism. W.H. Freeman, San Francisco (1982)
Riker, W.H.: The Art of Political Manipulation. Yale University Press, New Haven, CT (1986)
Robinson, I.S.: The Papacv 1073-1198. Continuitv and Innovation. Cambridge University Press, Cambridge (1990)
Edoardo, R.A.: Il principio maggioritario nella storia del Diritto Canonico, Archivio? Giuridico Filippo Serafini 4(IX), 15–G7 (1925)
Saari, D.G.: Geometry of Voting. Springer, Heidelberg (1994)
Sigmund, P.E.: Nicolaus of Cusa and medieval political thought. Harvard University Press, Cambridge, MA (1963)
Sigmund, P.E.: Nicolaus of Cusa: The Catholic Concordance. Cambridge University Press, Cambridge (1991)
Ullmann, W.: Law and Politics in the Middle Ages. In: An Introduction to the Sources of Medieval Political Ideas, Cambridge University Press, Cambridge (1975)
Vauchez, A. (dir.): Apogée de la Papauté et expansion de la Chrétienté. In: Mayeur, J.M., et al. (dir.) Histoire du Christianisme, vol. 5, pp. 1054–1274. Desclee, Paris (1990)

Static and Dynamic Coalition Formation in Group-Choice Decision Making

Tom Wanyama

Department of Networks
Makerere University
P.O. Box 7062
Kampala, Uganda
wanyama@cit.mak.ac.ug

Abstract. In Group-Choice Decision Making (GCDM) where a number of stakeholders are involved in choosing a single solution from a set of available solution options, it is common for the stakeholders to form coalition during negotiation in order to increase their individual welfare. It is also common to use Multi-Agent Systems (MAS) to automate GCDM processes. In such MAS, agents have to form coalitions like their human counterparts, and within each coalition, the individual agents behave according to the strategies of their clients. This paper presents a coalition formation engine that has two coalition formation algorithms. One of the algorithms is based on the concept of static coalition formation, and the other is based on the concept of dynamic coalition formation. Moreover, the coalition formation engine is coupled with algorithms that govern the social behavior of the agents in their coalitions, to form an agent negotiation engine. Finally, this paper presents an example and simulation results that illustrate the operational effectiveness of the two coalition formation algorithms, as well as the algorithms that govern the social behavior of the agents.

Keywords: Agent, Group-Choice, Coalition-formation, Negotiation, Static, and Dynamic.

1 Introduction

Coalition formation has the potential of increasing the welfare of negotiating entities. Therefore, the notion of forming coalitions is normally applied in various domains, including commerce, engineering and social planning. Furthermore, coalition formation in Multi-Agent Systems has attracted a lot of attention because agents can easily to be used to represent the negotiating entities, and they are capable of finding more profitable coalitions than humans in complex environments.

Although a variety of automatic coalition formation models have been proposed in literature, they all fall under two distinctive categories: the utility based models and the knowledge based models. Utility based models (e.g. Game Theoretic Models) normally aim at identifying stable coalitions of negotiating agents, as well as determining appropriate models for sharing coalition benefits [1, 2, 3, 8]. Most

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 45–56, 2007.

of these algorithms are naturally centralized; this is against the principle of decentralization, which is fundament to the concept of Multi-Agent Systems (MAS). The computational requirements of utility based coalition formation algorithms make them hard to implement [4]. Moreover, the algorithms normally implicitly assume that the agents surrender all their autonomy to their coalitions.

Knowledge based algorithms for coalition formation [5, 6, 7] depend on the skills of the individual agents to establish the most beneficial coalition structures. These algorithms are normally computationally feasible, but they invariably have the following shortfalls:

- They implicitly assume that there is no coalition structure that is more beneficial to all agents, than the grand (coalition of all negotiating agents) coalition [4].
- Like the utility based coalition formation algorithms, knowledge based algorithms normally assume total loyalty of agents to their coalitions.

Coalition formation algorithms can also be classified into static and dynamic algorithms. We define static coalition algorithms as those that do not allow for possible changes to the membership of coalitions as new information emerges. On the other hand, dynamic algorithms address the need for changes in the membership of coalitions as new information emerges [3]. While many coalition formation algorithms of both types (static and dynamic) have been proposed in literature, the issue of using multiple algorithms so to give agents the ability to benefit from the strength of both types of coalition formation algorithms has not attracted any attention. This paper presents an agent negotiation engine that can employ multiple coalition formation algorithms. The engine is made up of four main components: the first component contains the algorithms for coalition formation, the second component contains algorithms for the individual and group ranking of the solution options, the third component contains a model that represents the level of commitment of the agents to their coalitions, and the fourth component store the parameters for decision making. Besides increasing the utility of the negotiating agents through coalition formation, and controlling the dynamics of the negotiation process, our negotiation engine has the following advantages in respect of coalition formation:

- The negotiation engine takes into account the fact that although agents usually join coalitions to maximize the benefits associated with the goals that are share among the coalition members; within their coalitions, the individual agents act to maximize the benefits associated with the unshared goals. Our coalition formation model does not assume total loyalty of agents to their coalitions. Instead, it allows the agents to have commitment levels any where between the extremes of being absolutely loyal, and being entirely disloyal.
- Most knowledge based coalition formation models are applicable in environments where agents form a single grand coalition. However, a variety of coalition structures are normally feasible in practice. Our model is not limited to any type of coalition structures.
- The model does not add any extra communication requirements to the MAS.

The rest of this paper is arranged as follows, Section 2 discussed the coalition formation problem. In Section 3, we present our negotiation model, as well as its coalition formation algorithms. Section 4 deals with simulation experiments that illustrate the capabilities of the coalition formation model. Finally, conclusions are given in Section 5.

2 The Coalition Formation Problem

To develop satisfactory automatic coalition formation algorithms, it is necessary to understand the general coalition formation problem. The first step towards this is to partition the problem into small sub-problems that can be solved directly by executable algorithms. From our experience of designing and implementing a Group-Choice Decision Support System (GCDSS) based on multi-agent technology, we have identified the following coalition formation sub-problems to be of utmost importance:

1. Representation of when to join and/or leave a coalition.
2. How to handle the various feasible coalition structures.
3. Determination of what agents gain and/or loss when they join coalitions.

In the following paragraphs, we discuss each of the above coalition formation sub-problems in detail.

The general goal for coalition formation is maximizing utility, but the actual reasons for forming coalitions are normally different for different agents, and for different negotiations. Figure 1 shows the relationship between the reasons for forming coalitions and the goals of coalition formation. Furthermore, the Figure illustrates that the reasons for coalition formation fall in two categories. The first category usually result in dynamic coalition formation because the parameters upon which the decision to join or leave a coalition is based (e.g. similarity of preferences),

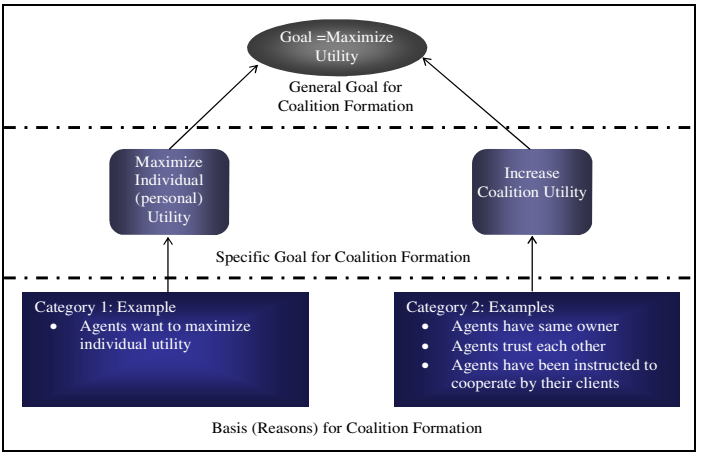


Fig. 1. Relationship between the Basis for Forming Coalition and the Goals of Coalition Formation

change as the negotiation process progresses, resulting in dynamic coalition structures. On the other hand, coalition formation reasons of the second category usually result in static coalition structures.

Negotiation in MAS can involve a variety of coalition structures. Therefore, practical coalition formation models must be able to operate with different coalition structures.

Figure 2 presents some examples of coalition structures that are briefly described the coalition structures in the Figure as follows:

- Figure 2 (a): All agents in the MAS belong to the same grand coalition.
- Figure 2 (b): Agents A1, A2, and A3 are in one coalition, and agents B1 and B2 are in another coalition.
- Figure 2 (c): Agents A1, A2, and A3 are in the same coalition, and they do not consider agents B1 and B2 to be in their allies. Yet, agents B1 and B2 consider A1, A2, and A3 to be their allies.

Figure 2 (d): Agent B1 is an ally of all the other agents in the MAS, but agent B2 is an ally of agent B1 only.

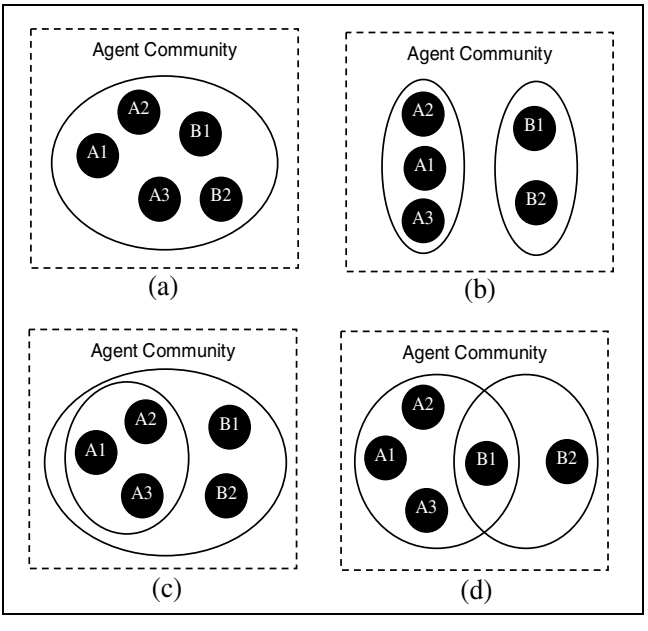


Fig. 2. Examples of Coalition Structures

Once an agent has joined a coalition, it is expected to compromise on some issues in order to gain on others. Consequently, coalition formation models must have representations of what agents are willing to gain and/or loss when they join coalitions. Such representations are related to the commitment levels of the agents to their coalitions. A totally committed agent is one that is willing to give up anything in

order to reach agreement with the other coalition members. On the other hand, a totally uncommitted agent is one that is unwilling to give up anything. Other levels of agent commitment lay between these two extremes. Models for representing what agents are willing to gain and/or loss when they join coalitions must be able to convert any agent commitment level into tangible negotiation tradeoffs.

3 Agent Negotiation Engine with Coalition Formation Capabilities

Our coalition formation algorithms are incorporated into an agent negotiation engine for collaborative decision-making called the **A**gent **N**egotiation **E**ngine for **Co**llaborative **De**cision **M**aking (ANE-CODEM). This engine is presented in detail in Wanyama and Far [9]. Figure 3 shows that the main components of the modified ANE-CODEM are the following: the Algorithm Component, the Social Component, and the Commitment Component.

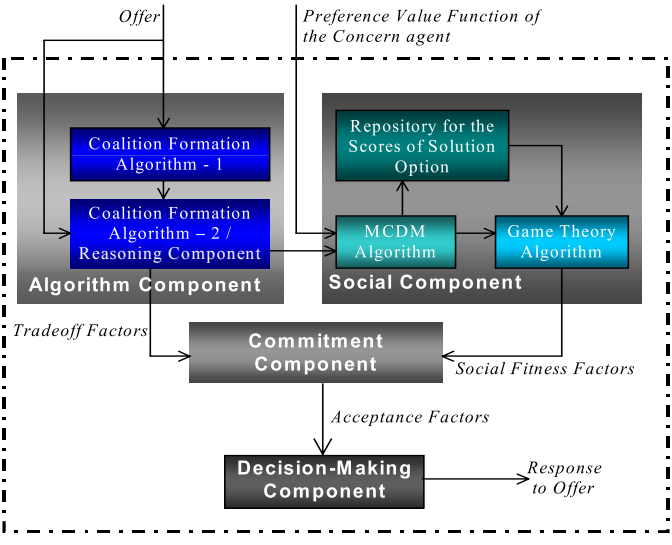


Fig. 3. Coalition Formation Model

Moreover, Figure 3 illustrates that the modified ANE-CODEM works as follows:

1. On receiving an offer, an agent sends it to the Algorithm Component of the negotiation engine to determine whether the proposer is an ally. If the proposer is not an ally, its offer is sent to the decision-making component so that the agent responds to the offer. On the other hand, if it is determined that the proposer of the offer is an ally then the information generated in the Algorithm Component is sent to the Social Component for further processing.

2. In the case of our Group-Choice Decision Support System (GC-DSS) based on multi-agent technology, the Algorithm Component estimates the preference value function of the proposer. This function is sent to the Social Component, which uses a MCDM model to determine the estimated scores of all the solution options for the proposer. The estimated scores of the solution options for all coalition members which have already made their offers, together with the scores of the solution options for the concern agents are fed in a Game theory model to generate the *Social Fitness Factors* (SF_f) of the Solutions. These factors represent the tradeoffs associated with the solution options in respect of the coalition.
3. Besides generating the estimated preference value function of the proposer, the Algorithm of component of our GC-DSS determines the *Tradeoff Factors* (T_f) of the solution options. These factors represent the tradeoffs associated with the solution options with respect to the concern agent. The *Tradeoff Factors* and the *Social Fitness Factors* of each of the solution options are combined in the Commitment Component to generate the *Acceptance factors* (A_f) of the solution options. The *Acceptance Factor* of a solution option is a measure of the ability of the solution to satisfy the combined preferences of the coalition members, from the perspective of a given coalition member. The generated Acceptance factors are used to update the offer acceptance/rejection criteria in the decision-making component of the agent before responding to the current offer. Furthermore, the *Acceptance factors* associated with the last offer in each negotiation round are used in the modification of the *preference value function* of the concern agent.

The coalition formation component of ANE-CODEM enables each agent to select individually its allies or coalition. Therefore, the component supports the formation of any coalition structure, and it does not require any extra communication or central coordination entity or agent. In the following subsections, we briefly describe the major components of the modified ANE-CODEM, in the context of coalition formation.

3.1 The Algorithm Component

The Algorithm Component of our coalition formation model can have multiple coalition formation algorithms. In view of that, the GC-DSS we developed has two coalition formation algorithms. The first algorithm is based on ownership (example of static coalition structures), while the second is based on similarity of preferences (example of dynamic coalition structures). The two algorithms can work in cascade where the ownership criterion is initially applied, and if not met, then the similarity of preferences is applied. Alternatively, one of the algorithms can be bypassed, leaving the Algorithm Component to have the characteristics of the remaining coalition formation algorithm. Besides the coalition formation algorithms, the Algorithm Component has an algorithm for determining the tradeoff factors of the solution options (see Figure 4).

3.1.1 The Ownership Based Coalition Formation Algorithms

This algorithm is very simple; all it requires is the agents to have the ability to identify the owners of their negotiation opponents. When agents make offers in the first negotiation around, they also broadcast their ownership identity. This enables each agent to identify the member of its coalition (i.e. the agents with the same owner as itself, and/or the agents owned by allies of its client). If agent is determined to be an ally (member of the coalition), its preference model is estimated using part of the coalition formation algorithms that is based on similarity of preferences. Thereafter, the model is sent to the social component for further processing.

3.1.2 The Coalition Formation Algorithms Based on Similarity of Preferences

This algorithm involves estimating the preference value function of the negotiation opponents of the concern agent based on the offers that they make, and comparing the estimated preference value functions of the opponents with the preference value function of the concern agent. Agents, whose preference models are found to be similar to that of the concern agent, are treated as allies of the agent, and their preferences are used in the social component to finally set the decision parameters in the decision component. Figure 4 presents the pseudo code of the coalition formation algorithms based on similarity of preferences. In the code, it is assumed that the concern agent prefers Solution Option B, and an offer of Solution Option A has been made by one of the negotiation opponents.

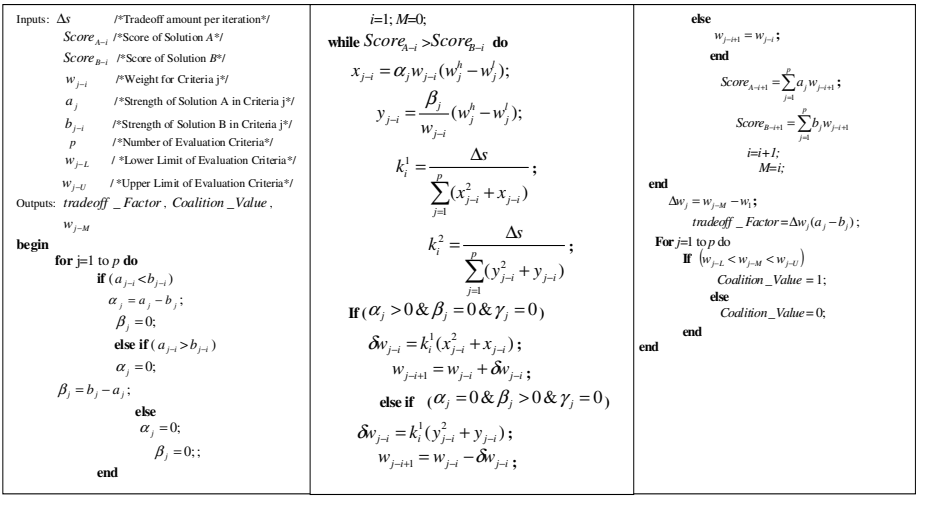


Fig. 4. Coalition Formation Algorithm based on the Similarity of Preferences

This algorithm is based on the concept of Qualitative Reasoning [11], and it works as follows:

1. The first 'For' loop determines the strength difference between solution option A and solution option B in each evaluation criteria.
2. The first two lines of the 'While" loop determine parameters which correspond to the reasoning that the difference between the utilities

associated with different solution options depends more of the evaluation criteria (decision variables) that have high criteria weights (preferences values) than on the criteria with low weights.

3. The 'If' loop within the 'While' loop determines the estimated criteria weights (preference values).
4. The last part of the 'While' loop reevaluates the solutions to check if the estimation of the preference values is working.
5. After the 'While' loop, the algorithm calculates the amount of tradeoff – factor associated with the offered solution option.
6. Finally, the algorithm compares the estimated preference value function of the proposer of the offer to that of the concern agent. If all the estimated preference values fall within the limits set by the agent client of determined by the agents its self, the *Coalition_Value* is set to 1, meaning that the proposer of the offer is an ally (member of the coalition). Otherwise, the *Coalition_Value* is set to 0, implying that the proposer is not an ally.

3.2 The Social Component

This component has an algorithm that determines the *Social Fitness Factors* of the solution options as follows:

- Based on the estimated preference value function of the proposer, estimate the scores of the solution options using a MCDM model.
- Using the scores of the solution options for all the negotiation opponents, which have so far made their offers, as well as the scores of the solution options for the concern agent as the input to a *Game Theory* model, determine the *Social Fitness Factors* of the solution options.

The Social Fitness Factors represent the tradeoff associated with each of the solution options with respect to the coalition of the concern agent. Therefore, if the concern agent is totally committed to its coalition, then the solution option with the best Social Fitness Factors would be an acceptable option if offered. Otherwise, the agent has to identify the solution option that meets its level of commitment to the coalition by integrating the Social Fitness Factors with the Tradeoff Factors.

3.3 The Commitment Component

Coalition formation is a result of shared goals that agents need to optimize through cooperation. Nevertheless, the individual agents normally have unshared goal that they need to optimize through their individual actions. The level an agent is concern about the shared goals (or the coalition tradeoffs) in comparison to its concern about the unshared goals (or the individual tradeoffs), is represented by the *Commitment Level* (ω) of the agent. In other words, the *Commitment Level* of an agent is a measure of the commitment of the agent to its coalition. The ability of a solution option to satisfy the commitment level of an agent is represented by the *Acceptance Factor* (A_f), which is given by Equation 1.

$$A_f = \omega.T_f + (1-\omega)SF_f \ . \quad (1)$$

where, $0 \leq \omega \leq 1$.

The Acceptance Factors of all the solution options are sent to the decision-making component of each agent to update its decision parameters. It should be noted that if an agent makes an offer, which has the best known *Acceptance Factor*, that offer is accepted even if the proposing agent is not a member of the coalition. Moreover, the Acceptance Factors generated at the end of the negotiation round are used as the basis for adjusting the preference value function of the concern agent

4 Example

We adopted a group-choice decision making problem from Wanyama and Far [10]. However, only the problem solving processes that are relevant to coalition formation are presented in this paper.

In the example, an organization dealing in chinese food intends to change its business model from delivering food to households based on phone orders, to building a webshop where orders can be placed. The team developing software for the webshop intends to use a COTS to provide most of the functional requirements, then develop the extra requirements and interfaces inhouse. It is assumed that the three outlets, which are to be connected to the web-based shop, have to contribute financially towards its development, deployment and maintenance. Therefore, the products are evaluated based on the quality and business concerns of the outlets managers. Four COTS products have been identified as possible solutions to the problem, and the selection of the best-fit solution potion is to be carried out online. Therefore, each outlet manager has been availed with an Agent that has automatic negotiation capabilities enabled the modified ANE-CODEM.

4.1 Simulation Experiment

Negotiation experiments were carried out, based on the preference value functions developed by three stakeholders through their respective agents *Agent a*, *Agent b*, and *Agent c*. Therefore, the experiments simulated the negotiation between the three stakeholders. During negotiation, the agents made offers in turns until an agreement was reached. Moreover, the agents were set to start negotiations with their commitment levels set to 0.5, and they were allowed to adjust this value whenever it was required to break deadlocks. In all the experiments, the utility for each agent associated with the offers made was noted. Figure 6 shows the utilities of two of the agents (*Agent a,* and *Agent b*) for the offer made during the negotiation processes for the six simulation experiments. In the Figure, the circles represent the offers made by *Agent b,* and the crosses represent the offers made by *Agent a*. A cross inside a circle represents the agreement offers.

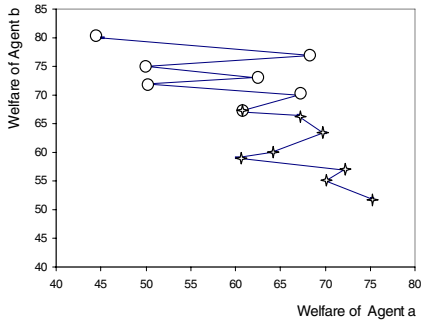
Since the main objective of the experiments was to determine the effects of different coalition formation algorithms, and/or coalition structures on the utility and dynamics of a negotiation process, each of simulation experiments were carried out with one of the following conditions:

- No coalition formation.
- *Agent b* and *Agent c* are allies (are in a static coalition) against *Agent a*, and *Agent a* does not consider them its allies.
- *Agent b* and *Agent c* are allies, and they do not consider *Agent a* to be in their ally. Yet, *Agent a* considers both *Agent b* and *Agent c* to be its allies. This coalition structure is static through out the negotiation.
- *Agent a*, *Agent b*, and *Agent c* are in one static grand coalition.
- *Agent b* and *Agent c* can form a coalition based on the similarity of their preferences, but no alliance with *Agent a* is permitted.
- Each of the three agents can form alliances with any other agent based on the similarity of their preferences.

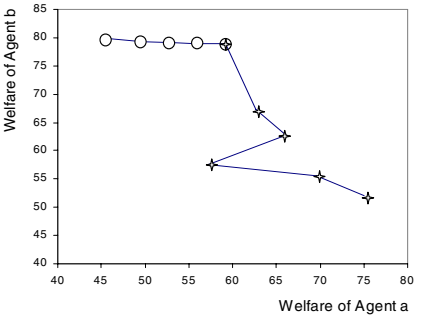
These negotiation condition were achieved by either bypassing one of the negotiation algorithms, or by operation the two algorithms in the "OR" or the "AND" mode. The OR-mode is where the proposer of an offer is required to meet the conditions of at least one of the algorithms to be a member of the coalition, and the AND-mode requires the proposer to satisfy the conditions of both coalition formation algorithms in order to be a member of the coalition.

4.2 Results

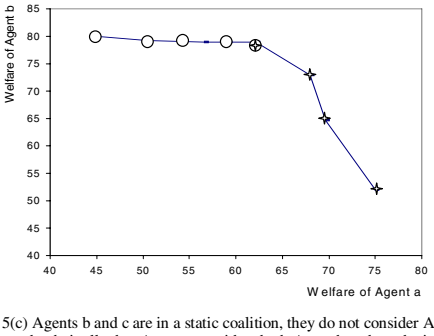
The results of the six simulation experiments are presented in Figure 5 below.



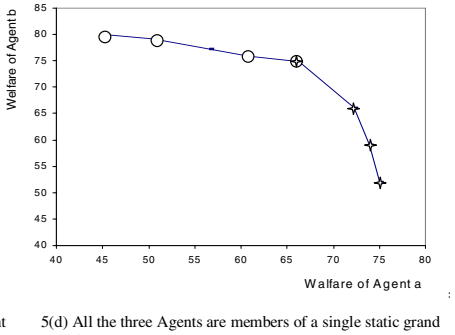
5(a) No Coalition Formation



5(b) Static coalition between Agent b and Agent c, Agent a acting

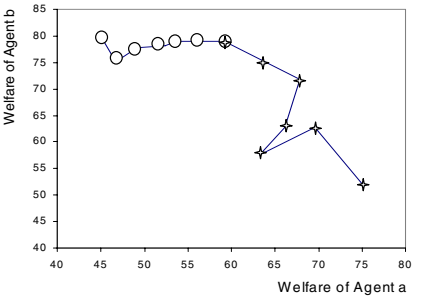


5(c) Agents b and c are in a static coalition, they do not consider Agent a to be their ally, but Agent a considers both Agent b and c to be its allies

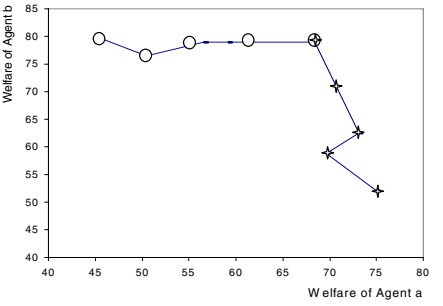


5(d) All the three Agents are members of a single static grand coalition

Fig. 5. Results of the Simulation Experiments



5(d) Agents b and c can form a coalition based of the similarity of their preferences, but no coalition with Agent a is permitted



5(f) All the three Agents are free to form alliances based on the similarity of their preferences

Fig. 5. (*continued*)

4.3 Discussion of Results

Generally, coalition formation creases utility. That is, the utility of the agreement offer for no coalition formation negotiation (Figure 5(a)) is smaller then the utility of agreement offers for any of the negotiations with coalition formation (Figures 5(b) - (f)). However, simply making alliances based on static parameters such ownership does not ensure maximizing utility (compare utility of agreement offer in Figures 5(d) and 5(f)). Therefore, the major issue of coalition formation in MAS is not the coalition structures themselves, but the search for the structures that maximize utility. This implies that concept of designing coalition formation algorithms that target particular coalition structures (e.g. the grand coalition), misses the main objective of coalition formation. Unfortunately, this is the approach that most knowledge coalition formation algorithms follow.

Figure 5 illustrates that coalition formation controls the dynamics of the negotiation process. Moreover, the Figure demonstrates that coalition formation expedites the negotiation process (forming coalitions resulted in few negotiation rounds), and agents negotiation in a smart manner, where they make offers that aim at identifying a mutually acceptable solution. Static coalition structures offer better control of the negotiation dynamics, and expedite the negotiation process more than the dynamic structures, but at the risk of reduced utility (compare Figures 5(c) and (d) and one hand, with Figures 5 (e) and (f) on the other).

5 Conclusions and Future Work

This paper deals with the issue of coalition formation in MAS; it relates the reasons for forming coalition to the goals of coalition formation, to the of coalition structure types and to the utilities of the coalition and of the individual agents. In addition, this paper presents a negotiation engine that is capable of employing multiple coalition formation algorithms. Currently, the engine has two coalition formation algorithms, one based of the ownership of the agents, and the other based on the similarities

between the preferences of the agents. Finally, the paper presents an example that illustrates the capabilities of our coalition formation algorithms.

For the future, this research will be extended in two major directions. Firstly, we would like to model coalition membership such that agents that belong to multiple coalitions can exhibit different levels commitment to different coalitions. Secondly, we would like to continue studying the coalition formation process in order to improve the efficiency of our coalition formation model.

References

1. Vauvert, G., Fallah-Segrouhni, A.E.: Coalition Formation among Strong Autonomous and Weak Rational Agents. In: Proc. 10th European Workshop Modelling Autonomous Agents in a Multi-agent World (MAAMAW), Springer, Heidelberg (2001)
2. Kahan, J.P., Rapoport, A.: Theories of Coalition Formation. Lawrence Erlbaum Associates, London (1984)
3. Klusch, M., Gerber, A.: Dynamic Coalition Formation among Rational Agents, IEEE Intelligent Systems, 42–47 (2002)
4. Jennings, N., Dang, V.D.: Coalition Formation Algorithms for Virtual Organisations, Intellingence, Agents, Multimedia Group, Electronics and Computer Sceince – University of Southampton (2004), Available online from: http://www.iam.ecs.soton.ac.uk/projects
5. Maheswaran, R.T., Basar, T.: Coalition Formation in Proportionally fair Division Auctions. In: The Autonomous Agents and Multi-Agent Systems Conference, Melbourne Australia, July 2003, AAMAS 2003
6. Abdallah, S., Lesser, V.: Organization-Based Coalition Formation. In: Kudenko, D., Kazakov, D., Alonso, E. (eds.) Adaptive Agents and Multi-Agent Systems II. LNCS (LNAI), vol. 3394, Springer, Heidelberg (2005)
7. Breban, S., Vassileva, J.: A Coalition Formation Mechanism Based on Inter-Agent Trust Relationships. In: AAMAS 2002, The Autonomous Agents and Multi-Agent Systems Conference, Bologna, Italy (July 2002)
8. Kraus, S.: Negotiation and Cooperation in Multi-agent Environments. J. Artificial Intelligence 94(1-2), 79–98 (1997)
9. Wanyama, T., Far, B.H.: An agent Negotiation Engine for Collaborative Decision Making. In: Torra, V., Narukawa, Y., Valls, A., Domingo-Ferrer, J. (eds.) MDAI 2006. LNCS (LNAI), vol. 3885, pp. 46–57. Springer, Heidelberg (2006)
10. Wanyama, T., Far, B.H.: Multi-Agent System for Group-Choice Negotiation and Decision Support. In: Kudenko, D., Kazakov, D., Alonso, E. (eds.) Adaptive Agents and Multi-Agent Systems II. LNCS (LNAI), vol. 3394, pp. 194–208. Springer, Heidelberg (2005)
11. Wanyama, T., Far, B.H.: Qualitative Reasoning Model for Tradeoff Analysis. In: Torra, V., Narukawa, Y., Miyamoto, S. (eds.) MDAI 2005. LNCS (LNAI), vol. 3558, pp. 99–109. Springer, Heidelberg (2005)

A Multicriteria Fuzzy System Using Residuated Implication Operators and Fuzzy Arithmetic

Sandra Sandri[1], Christophe Sibertin-Blanc[2], and Vicenç Torra[1]

[1] IIIA, Artificial Intelligence Research Institute CSIC, Spanish National Research Council
Campus UAB, 08193 Bellaterra, Catalonia, Spain
{sandri,vtorra}@iiia.csic.es
[2] IRIT, Université Toulouse 1, 21 Allées de Brienne, 31042 Toulouse Cedex, France
sibertin@univ-tlse1.fr

Abstract. We present a multicriteria fuzzy system using gradual rules and fuzzy arithmetic. We first present a multicriteria problem and its solution for the case of precise information. Then we extend the model to treat pieces of information that may involve imprecision/vagueness. We show that the use of residuated implication operators, employed by gradual rules, coupled with similarity relations offer a better treatment of the problem than a Mamdani-like approach.

1 Introduction

A crucial step for solving a decision making problem is to formulate the preferences of the individual. In a multicriteria decision making (MCDM) problem, such formulation should take into account the preferences of each of the considered criteria.

Several approaches have been defined to express the preferences among alternatives. The two main categories of such approaches are (i) assigning values to alternatives and (ii) defining preference relations between alternatives. In the first category, different methods consist of assigning different types of values. For example, real numbers [14,13], linguistic terms (in ordinal scales) [15], and fuzzy values [1]. Some methods [11] also permit the user to express preferences by means of nonhomogeneous types of values (numeric and linguistic). In the second category, methods also differ on the type of values used. For example, real values [2] or rough sets [10].

When preferences are expressed by assigning values to the alternatives, MCDM problems are often solved in a two stage process consisting of i) aggregation of the satisfaction of all criteria and (ii) ranking (or selection) of the alternatives according to the overall satisfaction. An alternative to this is (i) to build preferences from the satisfaction degrees and (ii) to aggregate the preferences.

In this paper, we are interested in the first category of problems. We consider the case in which rule bases are used to express the satisfaction of a user considering sets of criteria. The set of input variables in each rule base is a subset of the whole set of criteria considered by the user (possibly a singleton, i.e. a single criterium) and the output variable is the satisfaction obtained when only those criteria are considered. The overall satisfaction is obtained by merging together the satisfaction results yielded from each rule base using aggregation functions defined on the real line. Our aim in this paper

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 57–67, 2007.

is to propose a means to extend the solution to this kind of problem when all the values used in the rule bases are given in the form of imprecise or vague terms.

For that we consider that the rule bases are composed of gradual rules, of the type *"the more x is A, the more y is B"* [5]. We use similarity relations to overcome the inconsistency that is often yielded as a result of the use of this type of rules [8,9] . In what concerns the aggregation of the outcomes of the rules, we base our approach on the extension principle. The approach proposed in [1] is similar to ours in what regards aggregating fuzzy numbers, although in that case numbers are restricted to be triangular.

In the following, we present an example of the problem in the non-fuzzy case. We use this same example to illustrate the various issues that are raised as we treat the problem in the fuzzy case. On Section 3 we then give a series of basic definitions and notations. In Section 4 we propose an approach to extend the framework delineated in the example to the case of imprecise/fuzzy values and discuss some of its limitations. Finally, Section 5 brings the conclusion.

2 Running Example

Let us suppose a visitor asked us to rent a car for him based on the criteria $R = \{cost, power, age\}$. Let us suppose that his level of satisfaction in relation to each of these criteria, - named sat-, is obtained through the use of the set of rules given in Table 1, and expressed by values in the scale $< none, low, reasonable, high, perfect >$, and denoted by $< NS, LS, RS, HS, PS >$. The input values for $cost$ and $power$ are given in the scale $< low, medium, high >$ and denoted by $< L, M, H >$, whereas those for age are given in $< old, medium, new >$ and denoted by $< O, M, N >$. The rule "*high cost* $\rightarrow$ *no satisfaction*" is represented in the 3rd entry of the first KB.

Table 1. Knowledge bases for the car problem

$cost:$	L	M	H
$sat:$	PS	HS	NS

$power:$	L	M	H
$sat:$	LS	HS	RS

$age:$	O	M	N
$sat:$	LS	HS	PS

Let us suppose there exist 2 available cars $S = \{c_1, c_2\}$, such that c_1 is rather expensive, rather fast and rather old and c_2 is not very expensive, rather slow and rather new, and that they can be best modeled using only the values in $< L, M, H >$ as

- $cost(c_1) = H, power(c_1) = H, age(c_1) = O$
- $cost(c_2) = M, power(c_2) = L, age(c_2) = N$

Comparing the features of the cars with the knowledge in the rules we obtain

- $sat_{cost}(c_1) = NS$, $sat_{power}(c_1) = RS$, $sat_{age}(c_1) = LS$ and
- $sat_{cost}(c_2) = HS$, $sat_{power}(c_2) = LS$, $sat_{age}(c_2) = PS$.

The aggregation of the satisfactions using the minimum, the arithmetic mean (here approximated by the middle term) and the maximum are given in Table 2. In our case, with any of these operators, car c_2 would be preferred over c_1.

Table 2. Aggregated results for cars c_1 and c_2

	cost	power	age	min	mean	max
c_1 :	NS	RS	LS	NS	LS	RS
c_2 :	HS	LS	PS	LS	HS	PS

3 Basic Definitions and Notations

In this section we recall some basic definitions that are used in the rest of the paper and provide some notation. Most of the definitions and remarks are well-known in the literature.

In the rest of the paper, unless stated otherwise, we shall work with fuzzy subsets of the real line, so the domain U below is assumed to be $\mathbb{R}$. The core (respec. support) of a fuzzy set $A : U \to [0,1]$ is defined as $core(A) = \{x \mid A(x) = 1\}$ (respec. $supp(A) = \{x \mid A(x) > 0\}$). For any $\alpha \in [0,1]$, the α-cut of A is defined as $[A]_\alpha = \{x \in U \mid A(x) \geq \alpha\}$. A is said to be *normalized* when there exists x such that $A(x) = 1$, and *convex* when for all x, y, z, if $x \leq y \leq z$, $A(y) \geq min(A(x), A(z))$. A linear by parts convex fuzzy set A, a trapezoid, is denoted as $< a_1, a_2, a_3, a_4 >$ where $supp(A) = (a_1, a_4)$ and $core(A) = [a_2, a_3]$.

An operator $\top : [0,1]^2 \to [0,1]$ is called a *t-norm* if it is commutative, associative, monotonic and has 1 as neutral element. A *residuated implication operator* $\to_\top$ is defined as $a \to_\top b = \sup_{c \in [0,1]} \top(a,c) \leq b$, where $\top$ is a left-continuous t-norm ($\to_\top$ is said to be the residuum of $\top$). One well-known residuated operator is Gödel implication (residuum of $\top = \min$) defined as $a \to_{\top G} b = 1$ if $a \leq b$ and $a \to_{\top G} b = b$ otherwise. Another one is Goguen implication, defined as $a \to_{\top\Pi} b = 1$ if $a \leq b$ and $a \to_{\top\Pi} b = b/a$ otherwise. Also noteworthy is the so-called Rescher-Gaines implication function, defined as $a \to_{\top RG} b = 1$ if $a \leq b$ and $a \to_{\top RG} b = 0$ otherwise, which is not a residuated operator itself but is the point-wise infimum of all residuated implications.

Residuated implication operators are used to model *gradual fuzzy rules*, whose semantics corresponds to statements of the form *"the more x is A, the more y is B"* [5]. Once a (continuous) t-norm $\top$ is fixed, a gradual fuzzy rule "If x is A_i then y is B_i" induces a fuzzy relation between input and output values which is defined as $R_i(x,y) = (A_i \to_\top B_i)(x,y) = A_i(x) \to_\top B_i(y)$. Then, given a gradual rule set $K = \{$"If x is A_i then y is B_i"$\}_{i \in I}$, the induced global fuzzy relation R_K is the intersection of the individual ones, i.e. $R_K(x,y) = \min_{i \in I} R_i(x,y)$. Finally, given an (imprecise) input "x is A_0", the output "y is B_0" produced by K is usually computed as sup-$\top$ composition of A_0 with R_K, that is, $B_0(y) = output(K, A_0)(y) = \sup_x \top(A_0(x), R_K(x,y))$. In particular, if the input is a precise value, $x = x_0$, the expression is simplified to

$output(K, \{x_0\}) = R_K(x_0, y) = min_{i \in I} B'_i(y)$, where $B'_i(y) = \alpha_i \rightarrow_\top B_i(y)$, and $\alpha_i = A_i(x_0)$.

A similarity relation S on a domain U is a binary fuzzy relation, i.e. a mapping $S : U \times U \rightarrow [0, 1]$ that satisfies the following properties:

(i) $\forall x \in U, S(x, x) = 1$ (*reflexivity*);
(ii) $\forall x, y \in U, S(x, y) = S(y, x)$ (*symmetry*);

Some authors require similarity relations to also satisfy the T-norm transitivity property ($\top(S(x, y), S(y, z)) \leq S(x, z)$ for all $x, y, z \in U$ and some t-norm $\top$), but we do not take it into consideration here as it does not seem to play a role in our framework.

The application of a similarity relation S on a fuzzy term A, denoted by $S \circ A$, creates a "larger" term $approximately\_A$. Formally, we have

$$(S \circ A)(x) = \sup_{y \in U} \min(S(x, y), A(y)).$$

The set of similarity relations on a given domain U forms a lattice (not linearly ordered) with respect to the point-wise ordering (or fuzzy-set inclusion) relationship. The top of the lattice is the similarity $S_\top$ which makes all the elements in the domain maximally similar: $S_\top(x, y) = 1$ for all $x, y \in U$. The bottom of the lattice $S_\perp$ is the classical equality relation: $S_\perp(x, y) = 1$ if $x = y$, $S_\perp(x, y) = 0$, otherwise. The higher a similarity is placed in the lattice (i.e. the bigger are their values), the less discriminating it is.

Particularly useful are families of parametric similarity relations $\mathcal{S} = \{S_0, S_{+\infty}\} \cup \{S_\beta\}_{\beta \in I \subseteq (0, +\infty)}$ that are such that:

(i) $S_0 = S^\perp$,
(ii) $S_{+\infty} = S^\top$, and
(iii) $\beta < \beta'$, then $S_\beta \prec S_{\beta'}$.

Here $S \prec S'$ means $S(x, y) \leq S'(x, y)$ for all $x, y \in U$ and $S(x_0, y_0) < S'(x_0, y_0)$ for some $x_0, y_0 \in U$.

The family given by $S_\lambda(x, y) = \max(0, 1 - \lambda^{-1} \cdot |x - y|)$, where $\lambda > 0$, satisfies interesting properties such as compatibility with the Euclidean distance, preservation of convexity, order and core [3].

Given any function f from X to Y, the *extension principle* permits us to extend f to fuzzy sets. That is,

$$\hat{f}(A)(y) = \max_{x \in X, f(x) = y} A(x)$$

This expression is extended easily into functions $f : X_1, \ldots, X_N \rightarrow Y$. For illustration, when f is the sum, the extension principle permits us to compute the fuzzy sum of fuzzy sets A and B:

$$C_{A \oplus B}(y) = \sup_{(x_1, x_2)/x_1 + x_2 = y} \min(A(x_1), B(x_2))$$

4 The Proposed Fuzzy System

Let us suppose we have a set of criteria X to judge a set of objects S. We construct a set of fuzzy rule-based systems, such that each rule base KB_i deals with a set of input variables $x_i \subseteq X$ and an output variable sat_i defined on a domain Y.

Let us suppose that the rules in a given KB are given in the form

R_{ij} : If $x_i = A_{ij}$ then $sat_i = B_{ij}$

where A_{ij} is a fuzzy term defined in the domain of variable x_i and B_{ij} is a fuzzy term defined on Y. For simplicity sake, each rule base has a single input variable $x_i \in X$, but the model can be easily extended to deal with input variables that have to be taken into account conjointly.

In our example, the terms presented in Table 1 can be modeled by fuzzy terms. Let us suppose that the terms for the output variables are defined on the [0,1] domain and given by $NS :< 0, 0, .2, .4 >$, $LS :< .2, .4, .4, .5 >$, $RS :< .4, .5, .5, .6 >$, $HS :< .5, .6, .6, .8 >$, and $PS :< .6, .8, 1, 1 >$ (see Figure 1).

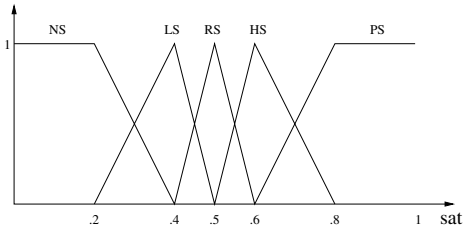


Fig. 1. Example of fuzzy sets partition for output variable sat in all KBs

To deal with the extended problem, we have to address two main issues:

1. what kind of inference should be used in the KBs and, accessorily, how the output of the various rules from the same KB should be aggregated together and
2. how the results yielded by the various KBs should be aggregated in order to produce an overall satisfaction value for each object.

To deal with the first issue we propose to use a true implication operator as the means to perform the inference in the KBs, along with a t-norm to aggregate the outputs concerning a single KB. To deal with the second issue we propose to apply the extension principle to aggregate the fuzzy sets yielded by the different KBs.

4.1 Inference

We propose to view the rules in the KBs as gradual rules, thus using residuated implication operator. This is done for two reasons:

– gradual rules seem to be closer to the way people express preferences (e.g. "the cheaper is the car, the better"), and
– the widely used Mamdani approach, in which the t-norm min is used as inference operator and max is used to aggregate the results, does not seem to be useful in this case.

Let us suppose that we use values for the input variables that resulted in the obtention of the terms in Figure 2.a (resp. Figure 2.b) for car c_1 (respec. c_2). In each figure, it is indicated the degree of compatibility of the fuzzy term in the rule premise (not illustrated here) and the input. For example, regarding the cost of car c_1, the input value is compatible with degree $\alpha = .75$ for the rule "*high cost* $\rightarrow$ *no satisfaction*", and degree $\alpha = .25$ for the rule with "*medium cost* $\rightarrow$ *high satisfaction*" (see Table 1).

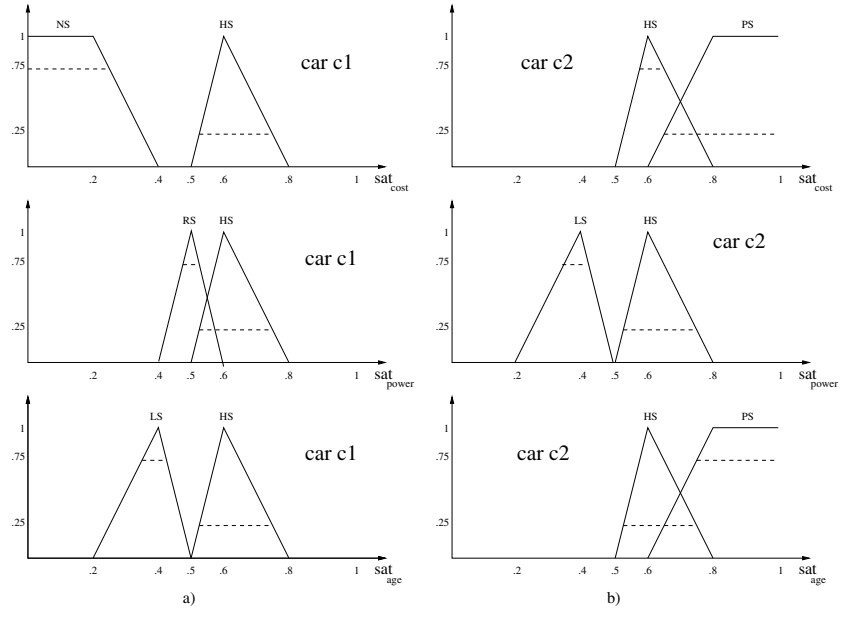


Fig. 2. Rule output fuzzy set and compatibility degree for cars c_1 (a) and c_2 (b) when their respective input is applied to the KBs

Figure 3.a) illustrates the application of the Mamdani framework (inference with min, rule aggregation with max) and the gradual rules one (inference with residuated implication operators Gödel and Goguen, aggregation with min), in KB_{cost} before the rule aggregation step. We see that none of the results is completely satisfactory.

The application of the max on the resulting output in Mamdani framework yields a non-convex, non-normalized result. The application of min in the gradual rule

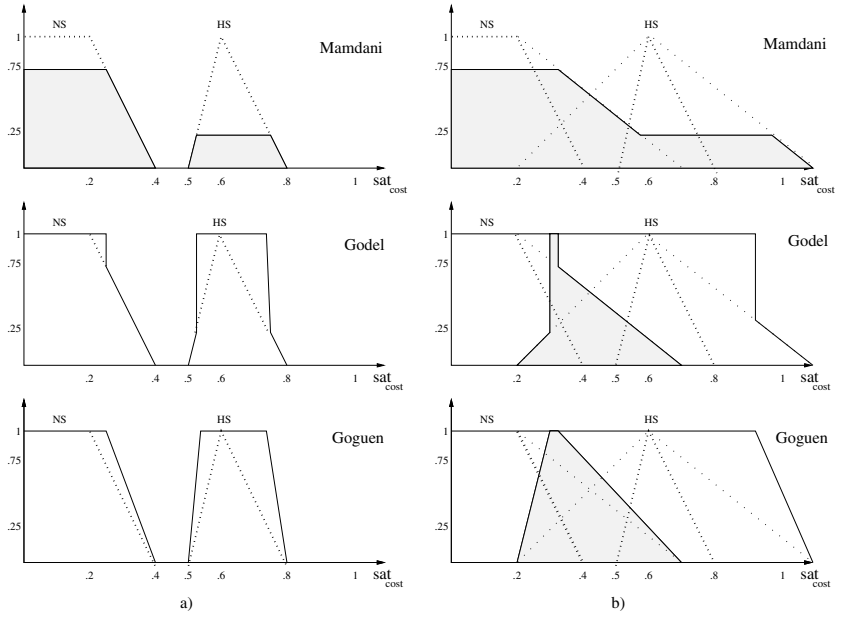


Fig. 3. Result of rule aggregation using different operators a) without similarity relation and b) with similarity relation $S_{.3}$

framework yields the empty fuzzy set as result, i.e., complete inconsistency, no matter which is the residuated implication operator used. Indeed, in the gradual rules framework, since the aggregation of the resulting fuzzy sets is done by a T-norm, complete inconsistency is very likely to occur.

Conditions characterizing consistency in systems of fuzzy gradual rules have been basically addressed by Dubois, Prade and Ughetto in [6,7]. A way to deal with inconsistency in gradual rules KBs is to apply a similarity relation [8,9] to the fuzzy terms in the output variable partition. A rule "If x is A then y is B" is then understood as "If x is A then y is $approximately\_B$".

Figure 3.b brings the result of applying the parameterized similarity relation family presented in Section 2 with $\lambda = .3$ on the rules output terms before inference. We see that we obtain good results with the gradual rules framework but that it is not appropriate for the Mamdani framework. In particular, it illustrates the fact that Goguen results, albeit less specific than Gödel ones, can be more easily represented, usually with only 2 level cuts.

Figure 4 brings the result of applying the same similarity relations family on the KB regarding power for car c_1 (see Figure 2.a) for the Goguen operator with parameters $\lambda = 0$, which is the equivalent of applying no similarity relation, and $\lambda = .3$. We see that in this case, the result with the application of S_0 produces better results than with $S_{.3}$.

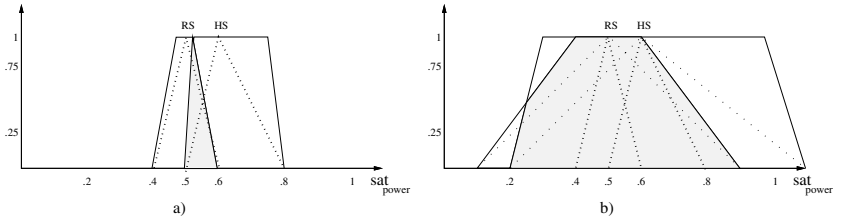


Fig. 4. Example of application of Goguen operator with a) S_0 and b) $S_{.3}$

To deal with inconsistency and yet obtain the most specific results, we adopt the approach proposed in [8] in which we use the smallest similarity relation that applied on the terms issues a normalized result upon the aggregation with min in the gradual rule framework.

Formally, for a precise input $x_i = a$, we just transform each rule R_{ij} : If $x_i = A_{ij}$ then $sat_i = B_{ij}$ into a rule $R_{ij}*$: If $x_i = A_{ij}$ then $sat_i = B_{ij}*$, where $B_{ij}* = B_{ij} \circ S_\lambda$, such that λ is the smallest parameter that makes $B_i*' = min_{i \in I} B_{ij}*' = 1$.

The results regarding the three KBs for each car in the gradual rule framework, using the smallest similarity relation capable of issuing a normalized result, are depicted in Figure 5 using Goguen operator.

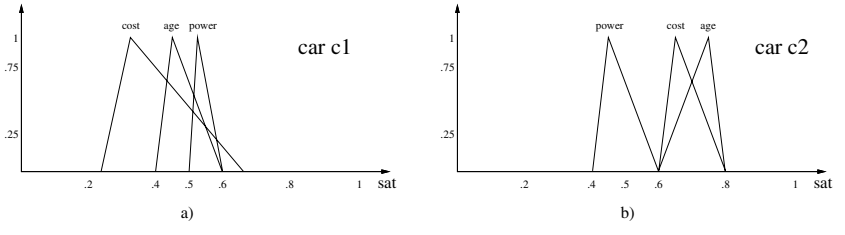


Fig. 5. Satisfaction results using Goguen implication and the most suitable similarity relation for cars a) c_1 and b) c_2

4.2 Satisfaction Aggregation

The inference in our framework is such that the satisfaction according to each criterium about a given object is modeled by fuzzy sets. Therefore, in order to aggregate them, the natural solution is to apply the extension principle on the same aggregating functions that would be used in a numerical non-fuzzy setting.

Given a set of results $sat_i(o) = B_i*'$, the final satisfaction concerning object o, and aggregation function f is calculated as

$$sat = B*' = \hat{f}(B_1*', ..., B_n*'),$$

where $\hat{f}$ is the fuzzy extension of function f.

Figure 6 illustrates the application of three different fuzzy operators obtained through the use of the extension principle: the min, the max and the arithmetic mean. We can see that the results are compatible with those obtained in the non-fuzzy case.

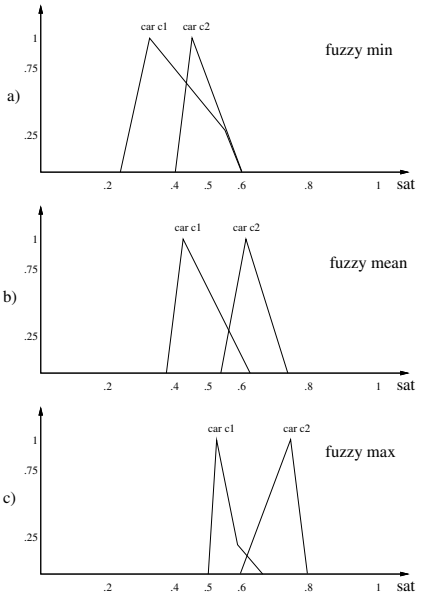


Fig. 6. Results of application of fuzzy operators: a) min, b) arithmetic mean, and c) max

4.3 Discussion

The equivalence between the non-fuzzy and fuzzy cases is only possible if the fuzzy sets are labeled in the same order as in the non-fuzzy case. But some other restrictions are needed to obtain the equivalence, mainly concerning how the fuzzy sets are defined in each variable domain, i.e. its partition.

Figure 7 brings examples of a variable partitions: 7.a) is a so-called Ruspini partition, in which $\forall x, \sum_i A_i(x) = 1$ (Figure 1 is also a Ruspini partition), 7.b) and 7.c) are partitions in which not all elements are covered, i.e. $\exists x, \sup_i A_i(x) = 0$ and 7.d) is a partition in which $\exists x, \sum_i A_i(x) \geq 1$, and for some elements there are more than 2 fuzzy sets with positive membership degrees for a given element of the variable domain.

Given a rule $A \rightarrow B$, if the input is a fuzzy set A' that is more specific than A, then the fuzzy set B' resulting from the inference step will be the same as B. More formally, due to the properties of residuated implication operators, it is straighforward to prove that if $\forall x, A'(x) \leq A(x)$, i.e. $A'(x)$ is included in $A(x)$ in the sense of Zadeh [16], then $B' = B$. Therefore, in what concerns inference, to obtain in the fuzzy case, values whose labels are *exactly* the same as those that we would obtain in the non-fuzzy case, the (conveniently ordered) fuzzy sets in the input partition need to have empty intersection, i.e. for any two fuzzy sets A_i and A_j in the input space such that $i \neq j$, we have to have $\forall x, min(A_i(x), A_j(x)) = 0$. Partitions in Figures 7.a) and 7.d) do not satisfy such a condition whereas 7.b) and 7.c) do.

In what regards aggregation, the condition above is not necessary for some operators, like min and max. On the other hand, it may be necessary but not suficient for some others operators. For instance, let us suppose that the output values regarding the

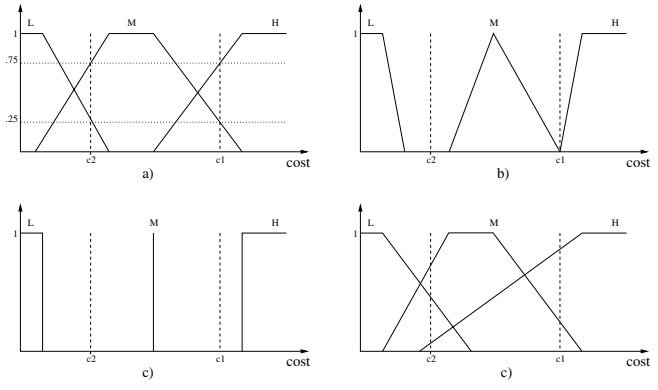


Fig. 7. Examples of fuzzy input partitions

satisfaction with respect to a given object in the non-fuzzy case are $B_1 = 1, B_2 = 2, B_3 = 3$. Now, let us suppose that the corresponding fuzzy sets are defined as $B_1 = < 0, 1, 1, 2 > B_2 =< 2, 2, 2, 2 > B_3 =< 2, 3, 3, 4 >$. Using the fuzzy arithmetic mean we obtain $B' =< 4/3, 2, 2, 8/3 >$ which is less specific than the original precise fuzzy set B_2.

To deal with uncovered domains, as in Figures 7.b) and 7.c), similarity relations can be applied also on input variables, as proposed in [9]. In this case, a rule 'If x is A then y is B" with uncovered regions in the input domain is understood as "If x is $approximately\_A$ then y is B" on those regions.

5 Conclusion

We presented here a multicriteria fuzzy system that extends the non fuzzy case in a natural manner. Inference is done using rule bases for each criterium, consisting of fuzzy gradual rules with Goguen residuated implication operator. The satisfaction with respect to to the criterium issued by its corresponding rule base is aggregated into a final satisfaction value by applying the extension principle on non-fuzzy aggregation operators (fuzzy arithmetic).

The extension principle can be burdensome to calculate. However, algorithms for many specific functions are quite straightforward when the fuzzy sets are LR [4], such as trapezes and Gaussian functions.

The approach shown here was inspired in social games [12], and we are working towards the integration of this model into SocLab (available at sourceforge.net). Sociology is mainly concerned with the kinds of links that make human beings to build societies in which they live, and social games study the links that social actors can produce, resulting in states of affairs where each actor accepts both his/her own position and the position of others. SocLab has been developed for games in which each actor behaves strategically although he has only bounded rationality capabilities. Each actor in a game controls a resource that is needed to some extend by at least one of the other actors. The action (move) performed by any given actor has an effect on all actors and

the satisfaction of each actor depends on the effects of the actions of all the actors. Our aim is to implement the approach described here in SocLAb, to help a system user to model a) the effect of each action on a given actor, and b) the overall satisfaction of the actor, which correspond respectively to the inference and satisfaction aggregation steps in our approach.

Acknowledgements. The authors thank anonymous reviewers for their comments and suggestions.

References

1. Chen, S.-M.: Fuzzy group decision making for evaluating the rate of aggregative risk in software development. Fuzzy Sets and Systems 118, 75–88 (2001)
2. Cheng, C.-H., Chang, J.-R.: MCDM aggregation model using situational ME-OWA and ME-OWGA operators. Int. J. of Unc. Fuzz. and Knowledge Based Systems 14(4), 421–443 (2006)
3. Drummond, I., Godo, L., Sandri, S.: Learning fuzzy systems with similarity relations. In: De Baets, B., Kaynak, O., Bilgiç, T. (eds.) IFSA 2003. LNCS, vol. 2715, pp. 516–523. Springer, Heidelberg (2003)
4. Dubois, D., Prade, H.: Possibility theory: an approach to computerized processing of uncertainty. Plenum Press, New York (1988)
5. Dubois, D., Prade, H.: What are fuzzy rules and how to use them. Fuzzy Sets and Systems 84, 169–185 (1996)
6. Dubois, D., Prade, H., Ughetto, L.: Coherence of Fuzzy Knowledge Bases. In: Proc. FuzzIEEE'96, New Orleans (USA), pp. 1858–1864 (1996)
7. Dubois, D., Prade, H., Ughetto, L.: Checking the coherence and redundancy of fuzzy knowledge bases. IEEE Trans. on Fuzzy Systems 5(3), 398–417 (1997)
8. Godo, L., Sandri, S.: A similarity based approach to deal with inconsistency in systems of fuzzy gradual rules. In: Proc. IPMU' 2002, Annecy (Fr), pp. 1655–1662 (2002)
9. Godo, L., Sandri, S.: Dealing with covering problems in fuzzy rule systems by similarity-based extrapolation. In: Proc. FuzzIEEE'02, Honolulu (USA) (2002)
10. Greco, S., Matarazzo, B., Slowinski, R.: Dominance-Based Rough Set Approach to Decision Involving Multiple Decision Makers. In: Greco, S., Hata, Y., Hirano, S., Inuiguchi, M., Miyamoto, S., Nguyen, H.S., Słowiński, R. (eds.) RSCTC 2006. LNCS (LNAI), vol. 4259, pp. 306–317. Springer, Heidelberg (2006)
11. Martinez, L., Liu, J., Yang, J.-B.: A fuzzy model for design evaluation based on multiple criteria analysis in engineering systems. Int. J. of Unc. Fuzz. and Knowledge Based Systems 14(3), 317–336 (2006)
12. Sibertin-Blanc, C., Amblard, F., Mailliard, M.: A coordination framework based on the Sociology of Organized Action. In: Boissier, O., Padget, J., Dignum, V., Lindemann, G., Matson, E., Ossowski, S., Sichman, J.S., Vázquez-Salceda, J. (eds.) Coordination, Organizations, Institutions, and Norms in Multi-Agent Systems. LNCS (LNAI), vol. 3913, pp. 3–17. Springer, Heidelberg (2006)
13. Smolíková, R., Wachowiak, M.P.: Aggregation operators for selection problems. Fuzzy Sets and Systems 131, 23–24 (2002)
14. Yager, R.R.: On ordered weighted averaging aggregation operators in multi-criteria decision making. IEEE Trans. on Systems, Man and Cybernetics 18, 183–190 (1988)
15. Yager, R.R.: An approach to ordinal decision making. Int. J. of Approx. Reasoning 12, 237–261 (1995)
16. Zadeh, L.A.: Fuzzy sets. Inform. and Control 8, 338–353 (1965)

A Behavioral Analysis in Decision Making Using Weather Information with the Fuzzy Target Based Decision Model

Akio Hiramatsu, Van-Nam Huynh, and Yoshiteru Nakamori

School of Knowledge Science
Japan Advanced Institute of Science and Technology
Nomi, Ishikawa, 923-1292, Japan
{akio-h,huynh,nakamori}@jaist.ac.jp

Abstract. In this paper, we discuss a behavioral model of decision making using weather information, making use of the so-called fuzzy target based decision model. Due to forecasting uncertainty in weather forecasts, many decision problems in practice influenced by weather information have been formulated as that of decision making under uncertainty. After introducing the fuzzy target based decision model which states that, after assessing a fuzzy target, the decision maker should select the decision which maximizes the probability of meeting his target, we will show that different behaviors of the decision maker about his target can lead to different decisions. This behavioral analysis not only provides an interpretation for influence of psychological personality features of the decision maker on his decisions, but also has a corresponding link to attitudes towards risk in terms of utility function.

1 Introduction

Many human activities and enterprizes are affected by uncontrollable future weather conditions. And the outcomes of many decision activities, ranging from fairly trivial (such as carrying an umbrella) to vitally important (substantially affecting the livelihoods of firms or individuals) are dependent on these weather conditions [14]. Therefore, it is natural that individuals/firms will seek out information about the future to make better decisions. The issue of using weather forecasting information in decision making has been a concern for many decades.

Due to forecasting uncertainty in weather forecasts and a wide range of services, it makes it difficult to utilize weather information for business purpose. For example, in Japan, beside Japan Meteorological Agency (JMA) there are also many private companies involving in weather forecasting services. However, in recent years, gross sales for private weather companies are depressed an annual total of thirty billion Japanese yen [12] and most of the certified weather forecasters have no opportunity to use their meteorological knowledge practically. To find a way out of these situations, it is said that something of highly sophisticated knowledge management, that means special observation equipment or original numerical weather prediction system, is useful [3].

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 68–79, 2007.

Recently, in an attempt of making use of weather forecasting information in decision analysis for transportation companies at Ishikawa (Japan) in winter season, Hiramatsu [7] has studied a knowledge management model for weather business (Fig. 1) and then applied the so-called cost/loss ratio model for some simple case studies [8]. Though the cost/loss ratio model has a long tradition of study, especially in the meteorological community, it is a very simple and may not be appropriate for complex problems.

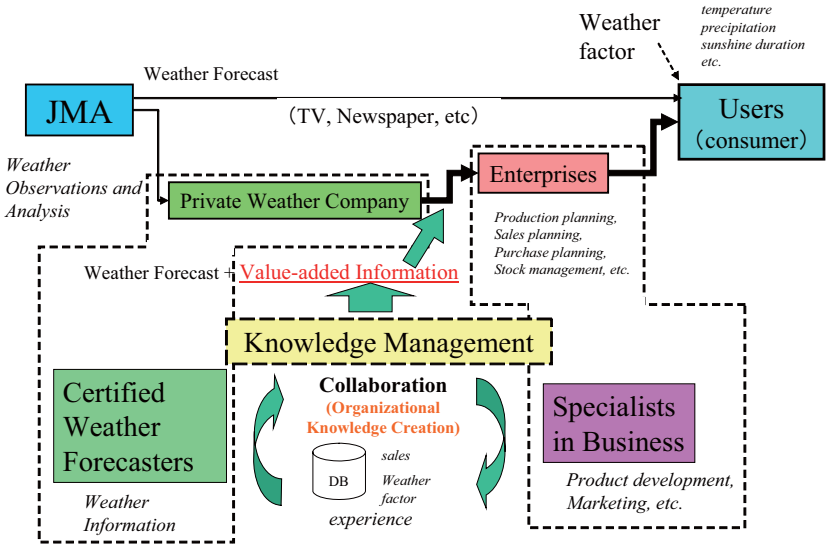


Fig. 1. The Concept of Knowledge Management for Weather Business

As mentioned in [14], a quite powerful treatment of weather-sensitive decisions made under uncertainty can be accomplished through the construction of the so-called decision analytic models. By this approach, many decision problems in practice influenced by weather information have been formulated as that of decision making under uncertainty (DUU) in the decision theory [18]. In addition, in [16] Thompson interestingly discussed the influence of behavioral/ability of the decision maker to the various use of decision strategies in meteorological decision making. On the other hand, the authors in [10,11] have recently studied a fuzzy target based decision model for DUU in which we can interestingly establish a direct link between the decision maker's different attitudes about target and different risk attitudes in terms of utility functions.

In this paper we will discuss the influence of behavior/ability of the decision maker in meteorological decision making using only the fuzzy target-based decision model but with various target attitudes. We will also show how our analysis is consistent with that conducted by Thompson [16]. The next section will briefly introduce the (fuzzy) target based decision model. In Section 3, after recalling

the behavioral analysis in meteorological decision making using different decision strategies by Thompson [16], a behavioral analysis using the fuzzy target based decision model is discussed. Section 4 presents some concluding remarks.

2 Target-Based Decision Model of DUU

In this section, we will introduce the fuzzy target based model for the problem of DUU, which has been recently investigated in [10,11]. Formally, the problem of decision making in the face of uncertainty is generally described using the decision matrix shown in Table 1. In this matrix, $A_i (i = 1, \ldots, n)$ represent the actions available to a decision maker, one of which must be selected. The elements $S_j (j = 1, \ldots, m)$ correspond to the possible values associated with the so-called state of nature S. Each element c_{ij} of the matrix is the payoff the decision maker receives if action A_i is selected and state S_j occurs. The uncertainty associated with this problem is generally a result of the fact that the value of S is unknown before the decision maker must choose an action A_i. Let us consider the decision problem as described in Table 1 with assuming a probability distribution $P_{\mathcal{S}}$ over $\mathcal{S} = \{S_1, \ldots, S_m\}$. Here, we restrict ourselves to a bounded domain of the payoff variable that $D = [c^{\min}, c^{\max}]$.

Table 1. Decision Matrix

Actions	State of Nature			
	S_1	S_2	$\ldots$	S_m
A_1	c_{11}	c_{12}	$\ldots$	c_{1m}
A_2	c_{21}	c_{22}	$\ldots$	c_{2m}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
A_n	c_{n1}	c_{n2}	$\ldots$	c_{nm}

As is well-known, the most commonly used method for valuating acts A_i to solve the DUU problem described by Table 1 is to use the expected utility value:

$$v(A_i) \triangleq EU_i = \sum_{j=1}^{m} P_{\mathcal{S}}(S_j) U(c_{ij}) \tag{1}$$

where U is a utility function defined over D.

On the other hand, each action A_i can be formally considered as a random payoff having the probability distribution P_i defined, with an abuse of notation, as follows:

$$P_i(A_i = x) = P_{\mathcal{S}}(\{S_j : c_{ij} = x\}) \tag{2}$$

Then, the target-based decision model [1] suggests using the following value function

$$\begin{aligned} v(A_i) &\triangleq P(A_i \geq T) \\ &= \sum_x P(x \geq T) P_i(A_i = x) \\ &= \sum_{j=1}^{m} P_{\mathcal{S}}(S_j) P(c_{ij} \geq T) \end{aligned} \tag{3}$$

where the random target T is stochastically independent of any random payoffs A_i, and $P(x \geq T)$ is the cumulative distribution function of the target T.

Interestingly, these two different procedures are shown to be both mathematically and observationally equivalent [1,4]. In other words, despite the differences in approach and interpretation, both target-based procedure and utility-based procedure essentially lead to only one basic model for decision making. It should be worth, however, emphasizing that while both target-based and utility-based decision making demand an understanding of probabilities, the utility-based model additionally requires a comprehension of cardinal utilities [2].

2.1 The Fuzzy Target Based Model

In some practical situations, due to a lack of information or statistical evidence, it may be difficult or impossible for decision maker to assess a probabilistic target as in the target-based decision model above. Instead, decision maker may be able to assess a fuzzy target based on his experience/feelings. It would be also noting that defining fuzzy targets is much easier and intuitively natural than directly defining random targets, especially in decision situations where the decision may be strongly influenced by the personal behavior of decision maker.

Under such observation, a fuzzy target-based decision model has been proposed and studied in [10,11]. Particularly, let us turn back to the DUU problem described in Table 1. In the fuzzy target-based decision model, we assume that the decision maker is able to establish a fuzzy target T which reflects his attitude. Then, after assessing the target he would select the course of action which maximizes the expected probability of meeting the target defined by

$$v(A_i) = \sum_{j=1}^{m} P_{\mathcal{S}}(S_j) \mathbb{P}(c_{ij} \geq T) \tag{4}$$

where $\mathbb{P}(c_{ij} \geq T)$ is a formal notation indicating the *probability of meeting the target* of value c_{ij} or, equivalently, the utility $U(c_{ij}) \triangleq \mathbb{P}(c_{ij} \geq T)$ in the utility-based language. In [9], two methods for calculating $\mathbb{P}(c_{ij} \geq T)$ from fuzzy targets have been discussed. We now briefly recall these two methods.

2.2 Probability of Meeting a Fuzzy Target

Note here that by a fuzzy target we mean a possibility variable T over the domain D represented by a possibility distribution $\mu_T : D \to [0, 1]$. For simplicity, we

assume further that T is normal, convex and has a piecewise continuous function with $\text{supp}(T) = [c^{\min}, c^{\max}]$, where $\text{supp}(T)$ denotes the support of T.

Simple Normalization. A direct and simple way to define $\mathbb{P}(c_{ij} \geq T)$ is making use of Yager's method [17] for converting a possibility distribution into an associated probability distribution via the simple normalization. Particularly, the possibility distribution μ_T of the target T is first converted into its associated probability distribution, denoted by P_T, as follows

$$P_T(t) = \frac{\mu_T(t)}{\int_{c_{min}}^{c_{max}} \mu_T(t)dt}$$

Then $\mathbb{P}(c_{ij} \geq T)$ is defined as the cumulative distribution function as usual by

$$\mathbb{P}(c_{ij} \geq T) = \int_{c_{min}}^{c_{ij}} P_T(t)dt \tag{5}$$

The α-Cut Based Method. Another method for inducing the utility function associated with $\mathbb{P}(c_{ij} \geq T)$ is based on the α-cut representation of fuzzy target T, where each α-cut T_α is viewed as a crisp approximation of T at the level $\alpha \in (0, 1]$. In particular, under an interpretation of $\{T_\alpha\}_{\alpha\in(0,1]}$ as a uniformly distributed random set, consisting of the Lebesgue probability measure on $[0, 1]$ and the set-valued mapping $\alpha \mapsto T_\alpha$ [5], the probability $\mathbb{P}(c_{ij} \geq T)$ is defined as

$$\mathbb{P}(c_{ij} \geq T) = \int_0^1 P(c_{ij} \geq T_\alpha)d\alpha \tag{6}$$

where $P(c_{ij} \geq T_\alpha)$ is the cumulative distribution function of the random variable having a uniform distribution over T_α. As mentioned in [9], the fuzzy target-based decision model using the α-cut based method for inducing the utility function in (6) reflects a stronger decision attitude towards the target than that using the simple normalization as in (5). So in this paper the α-cut based method is used for computing $\mathbb{P}(c_{ij} \geq T)$.

3 Behavioral Analysis Using Fuzzy Targets in Meteorological Decision Making

The problem of selecting an appropriate decision strategy in the face of forecasting uncertainty in weather information has been of concern to the meteorological community for many decades. In [6], Gleeson did describe two decision strategies based on the theory of games for this problem and simultaneously provided examples of their economic consequences. As mentioned in Thompson [16], there was a conceptual parallel between the various decision strategies and the familiar meteorological models which, however, might not be clearly evident to meteorologists. At the same time, the author did draw attention to this similarity and

interestingly discussed the influence of behavioral/ability of the decision maker to the various use of decision strategies via a simple example as shown in Table 2. In the following, we first introduce Thompson's analysis in meteorological decision making using three decision strategies, and then present a similar analysis by using only the fuzzy target-based decision model but with various target attitudes.

3.1 Three Decision Models and Thompson's Discussion

In Table 2, the first three rows (D_i) are the operational decisions which are related to the occurrence of weather events represented by columns headings (W_j). Elements (a_{ij}) in the table are gains and losses associated with corresponding pairs (D_i, W_j), i.e. a_{ij} is the profit or loss the decision maker receives if the decision D_i is taken and the weather event W_j occurs. The lower portion of the table shows the relative frequencies (p_j) of the occurrence of W_j, and the upper (p_j^{up}) and lower (p_j^{lo}) confidence limits for these frequencies.

Table 2. Gain/Loss Matrix with Various Weather Events [16]

		Weather Events			
		W_1	W_2	W_3	W_4
Decisions	D_1	1	4	1	0
	D_2	-1	0	4	1
	D_3	-3	0	2	7
Probability Information	p_j	0.10	0.25	0.50	0.15
	p_j^{up}	0.30	0.40	0.60	0.30
	p_j^{lo}	0.05	0.20	0.20	0

As for the purpose of selecting the proper decision D_i, the following decision models were considered according to different behaviors of the operator [16]:

1. If the operator wishes to conduct his operation so as to minimize large losses, he may select the course of action, denoted by $D_*^{(\min)}$, which will produce the maximum benefit from the *minimum economic expectation model* defined as

$$D_*^{(\min)} = \arg\max_{D_i} \sum_{j=1}^{m} a_{ij} p_{ij}^{(\min)} \tag{7}$$

where $p_{ij}^{(\min)}$ is a fictitious relative frequency of event W_j which is a maximum or minimum when a_{ij} is a minimum or maximum, respectively, subject to the constraint

$$p_j^{lo} \leq p_{ij}^{(\min)} \leq p_j^{up} \tag{8}$$

2. If the operator is risk neutrality, he may use the *mean economic expectation model*[1] to select course of action $D_*^{(\text{mean})}$ as follows:

$$D_*^{(\text{mean})} = \arg\max_{D_i} \sum_{j=1}^{m} a_{ij} p_j \tag{9}$$

3. If the operator is willing to risk possible large losses, he may decide on the course of action, denoted by $D_*^{(\text{max})}$, which will produce the maximim profit from the *maximum economic expectation model* defined by

$$D_*^{(\text{max})} = \arg\max_{D_i} \sum_{j=1}^{m} a_{ij} p_{ij}^{(\text{max})} \tag{10}$$

where $p_{ij}^{(\text{max})}$ is a fictitious relative frequency of event W_j which is a maximum or minimum when a_{ij} is a maximum or minimum, respectively, subject to the same constraint as $p_{ij}^{(\text{min})}$ in (8).

Applying these three models to the decision matrix given in Table 2 yields the results as shown in Table 3. For the detail of computations involved in this example could be referred to [6]. As we have seen, depending on the decision

Table 3. Economic Expectation with Various Decision Models [16]

		Economic Expectation		
		Minimum	Mean	Maximum
Decisions	D_1	*1.30*	1.60	2.20
	D_2	0.60	*2.05*	2.50
	D_3	-0.30	1.75	*2.85*

model used, different courses of action will be selected with different maximum gains produced. Now the question arises: which model should be used for making the decision? Clearly, no single course of action is appropriate in all cases and as Thompson [16] mentioned, the decision maker should assess the particular nature of the operation in order to determine an appropriate decision model. Simultaneously, Thompson offered several following suggestions. If the operator is working on a close economic margin, such an initial loss or series of losses, he would preclude continuation of the operation, then the minimum economic expectation model would be an appropriate one. However, if the operator has sufficient capital to absorb potential losses, then the mean or even the maximum, economic expectation model may be considered. Furthermore, it was also noticed that the personal philosophy of the decision maker may influence, to some extent, his selection of a decision model used. Particularly, a cautious operator may feel

[1] The expected value model.

more comfortable with a decision based on the minimum economic expectation, while a more adventurous entrepreneur may wish to gamble so that the maximum economic expectation model would be used.

In the above discussion of three decision models, it is required to have additional information such as upper and lower limits for probabilities of weather events, which may be practically difficult or expensive to be assessed. It is worth noting here that Thompson's analysis of the decision maker's attitudes towards risk is essentially based on the various use of these three models, but not in terms of 'utility' function as in theory of decision analysis with uncertainty. In the following we provide a behavioral analysis of the decision maker by means of the fuzzy target based decision model.

3.2 Fuzzy Targets and Behavioral Analysis

Let us consider to apply the fuzzy target based decision model to the problem of decision making using weather information as discussed above. Here we assume to have the relative frequencies (p_j) of the occurrence of weather event W_j only. Furthermore, we also suppose that depending upon the ability/behavior of the decision maker, he may first assess a fuzzy target T as his aspiration of profit, which can be defined as membership function $\mu_T : [a^{\min}, a^{\max}] \to [0, 1]$, where $a^{\min} = \min\{a_{ij}\}$ and $a^{\max} = \max\{a_{ij}\}$. Then he may select the course of action which maximizes the probability of meeting his target as follows:

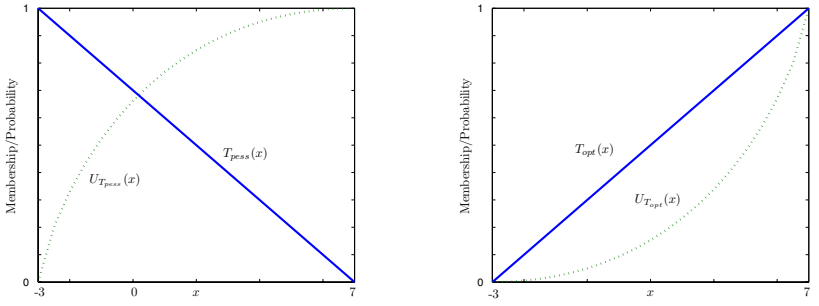
$$D_*^T = \arg\max_{D_i} \sum_{j=1}^{m} p_j \mathbb{P}(a_{ij} \geq T) \tag{11}$$

In this model, depending upon the ability/behavior of the decision maker he may assess his own fuzzy target, and several prototypical targets may be described as follows.

1. The first target is called the *pessimistic target*, which may correspond to the operator who wishes to avoid a serious loss and believes bad things may happen. Therefore he may have a conservative assessment of the target, which corresponds to ascribing high possibility to the uncertain target being a low gain or large loss. For simplicity, the membership function of this target is defined by

$$T_{pess}(x) = \begin{cases} \frac{a^{\max} - x}{a^{\max} - a^{\min}}, & \text{if } a^{\min} \leq x \leq a^{\max} \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

   Fig. 2 a) graphically depicts the membership function $T_{pess}(x)$ of this target and its corresponding utility function $U_{T_{pess}}(x) = \mathbb{P}(a_{ij} \geq T_{pess})$.
2. The second target expresses a *neutral* behavior on target of the decision maker and is represented by the possibility distribution $T_{neutral}(x) = 1$ for



a) Pessimistic target b) Optimistic target

Fig. 2.

$a^{\min} \leq x \leq a^{\max}$, and $T_{neutral}(x) = 0$ otherwise. In this case, it is easily to see that the model (13) becomes:

$$D_*^{neutral} = \arg\max_{D_i} \sum_{j=1}^{m} p_j \frac{a_{ij} - a^{\min}}{a^{\max} - a^{\min}} \tag{13}$$

which is equivalent to the mean economic expectation model as in (9).

3. The third one is called the *optimistic target*. This target would be set by the decision maker who is able to accept a risk of getting large losses, for example in case he has enough capital to absorb potential losses, and has an aspiration towards the maximal payoff. Formally, the optimistic fuzzy target, denoted by T_{opt}, can be defined as follows

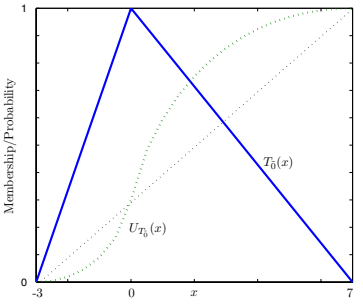
$$T_{opt}(x) = \begin{cases} \frac{x - a^{\min}}{a^{\max} - a^{\min}}, & \text{if } a^{\min} \leq x \leq a^{\max} \\ 0, & \text{otherwise} \end{cases} \tag{14}$$

Fig. 2 b) graphically depicts the membership function $T_{opt}(x)$ and the utility function $U_{T_{opt}}(x) = \mathbb{P}(a_{ij} \geq T_{opt})$ corresponding to this target.

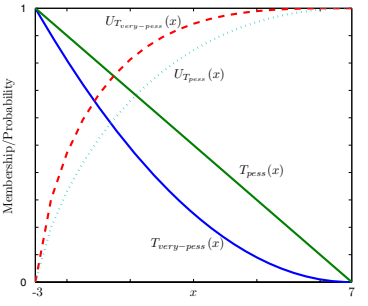
4. Interestingly, in some cases if the operator does not like any extremal views of target, he may express his aspiration level of gains or his acceptable level of losses, which is linguistically represented as "*about* a_0" having the membership function defined as

$$T_{\widetilde{a_0}}(x) = \begin{cases} \frac{x - a^{\min}}{a_0 - a^{\min}}, & a^{\min} \leq x \leq a_0 \\ \frac{a^{\max} - x}{a^{\max} - a_0}, & a_0 \leq x \leq a^{\max} \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

where $a^{\min} < a_0 < a^{\max}$, and this fuzzy target is called *unimodal*. For example, in the example above, if the decision maker taking $a_0 = 0$ and is graphically illustrated in Fig. 3 a), along with its corresponding utility function $U_{T_{\widetilde{a_0}}}(x) = \mathbb{P}(a_{ij} \geq T_{\widetilde{a_0}})$.



a) Unimodal target $T_{\tilde{0}}$



b) Very Pessimistic Target

Fig. 3.

Now let us turn back to the decision problem as described by Table 2. Applying the fuzzy target based decision model with different targets discussed above produces results as shown in Table 4.

Table 4. Expectation of Meeting the Target with Different Targets

		Expectation of Meeting the Target				
		Pessimistic	Neutral	Optimistic	About 0	Very Pessimistic
Decisions	D_1	0.7968	0.4600	0.1481	0.6330	*0.9053*
	D_2	*0.8075*	*0.5050*	0.1979	*0.6353*	0.9003
	D_3	0.7388	0.4750	*0.2392*	0.6005	0.8249

As we have seen in Table 4, the result interestingly reflects very well how a course of action selected is influenced by the behavior of the operator. That is, if the decision maker assessed a neutral target, the decision D_2 is selected as in the expected value model. If the decision maker wishes to get profit as big as possible, accepting a risk that if the desirable weather event will not occur, he may get a large loss, then he probably assesses an optimistic target which corresponds to D_3 being selected. In the case of the decision maker who assesses a pessimistic target, though D_2 is still selected, D_1 becomes more preferred over D_3. This means that the pessimistic operator is looking for sure of avoiding losses. However, by assessing a linear membership function for T_{pess} he is not enough pessimistic (see, Fig. 3 b)) to refuse D_2 which provides very attractive profit potential with relatively little risk. Let us assume that a more pessimistic target is assessed as follows:

$$T_{very-pess}(x) = \begin{cases} \frac{(a^{\max}-x)^2}{(a^{\max}-a^{\min})^2}, & \text{if } a^{\min} \le x \le a^{\max} \\ 0, & \text{otherwise} \end{cases} \tag{16}$$

Then the result corresponding this target is shown in the last column of Table 4, which suggests D_1 should be selected for surely avoiding a loss. It would be of interest to note that the nature of the target assessment may be also influenced by the personal philosophy of the decision maker. That is, a cautious decision maker may have a pessimistic-oriented target, while a more adventurous entrepreneur may prefer a optimistic-oriented target. However, unimodal targets would intuitively be the most natural to be assessed.

Now let us discuss a link between different attitudes about target and different attitudes towards risk in terms of utility functions. Looking at Fig. 2 a) and Fig. 3 b), we see that the pessimistic-oriented targets T_{pess} and $T_{very-pess}$ lead to concave utility functions and therefore, exhibit a risk-aversion behavior. By contrast, Fig. 2 b) shows that the optimistic target induces a concave utility function and thus equivalently corresponds to risk-seeking behavior. From the results shown in Table 3 and Table 4, we can see that the behavioral analysis by the fuzzy target based decision model is consistent with that discussed by Thompson [16] using three decision models. However, in our opinion, using fuzzy targets would be more nature and easier, while having a direct link to the traditional notion of utility functions. In addition, the unimodal target induces a utility function that is equivalent to the S-shape utility function of Kahneman and Tversky's prospect theory [13], according to which people tend to be risk averse over gains and risk seeking over losses. In the fuzzy target-based language, as the decision maker assesses his uncertain target as distributed around the modal value, he feels loss (respectively, gain) over payoff values that are coded as negative (respectively, positive) changes with respect to the modal value. This would lead to the behavior consistent with that described in the prospect theory and it is also suggested that this sort of target be the most natural one to occur.

4 Conclusion

In this paper, we have discussed a behavioral model of decision making using weather information by using the fuzzy target based decision model. It is observed from the comparative results that the behavioral analysis by the fuzzy target based decision model is not only consistent with that discussed in the literature of meteorological decision making, but also has a direct link to the well-known notion in the decision theory of attitudes towards risk in terms of utility functions. As for future work, we are conducting case studies for application of the fuzzy target based decision model to the problem of decision making using weather forecasting services of transportation companies in Ishikawa, particularly in the winter season.

Acknowledgements

The constructive comments from anonymous reviewers are greatly appreciated.

References

1. Bordley, R., LiCalzi, M.: Decision analysis using targets instead of utility functions. Decisions in Economics and Finance 23(1), 53–74 (2000)
2. Bordley, R.: Foundations of target-based decision theory. In: Brachinger, H.W., Monney, P.A. (eds.) Decision Analysis. From: Encyclopedia of Life Support Systems (EOLSS), Eolss Publishers, Oxford (2002)
3. Business and Weather Information Editing Committee (Ed.). Business and Weather Information – Report in the Front Line. Tokyo: Tokyo-do publishing 2004 (in Japanese)
4. Castagnoli, E., LiCalzi, M.: Expected utility without utility. Theory and Decision 41(3), 281–301 (1996)
5. Dubois, D., Prade, H.: Fuzzy sets, possibility and measurement. EJOR 40(2), 135–154 (1989)
6. Gleeson, T.A.: A prediction and decision method for applied meteorology and climatology, based partly on the theory of games. Journal of Atmospheric Sciences 17(2), 116–121 (1960)
7. Hiramatsu, A.: The Knowledge Management of Weather Information Business – A Study for Creation of Value Added Weather Information, Master's Thesis, JAIST: Ishikawa, Japan 2005 (in Japanese)
8. Hiramatsu, A., Nakamori, Y.: A study on decision-making models using long-range weather forecasting services. In: Proceedings of KSS 2006, Beijing, China, pp. 232–237 (2006)
9. Huynh, V.N., Nakamori, Y., Ho, T.B.: Utility function induced by fuzzy target in probabilistic decision making. In: Greco, S., Hata, Y., Hirano, S., Inuiguchi, M., Miyamoto, S., Nguyen, H.S., Słowiński, R. (eds.) RSCTC 2006. LNCS (LNAI), vol. 4259, pp. 296–305. Springer, Heidelberg (2006)
10. Huynh, V.N., Nakamori, Y., Ryoke, M., Ho, T.B.: A fuzzy target based model for decision making under uncertainty. In: IEEE International Conference on Fuzzy Systems, Vancouver, Canada, pp. 1611–1618 (2006)
11. Huynh, V.N., Nakamori, Y., Lawry, J.: A probability-based approach to comparison of fuzzy numbers and applications to target-oriented decision making, IEEE Transactions on Fuzzy Systems (to appear)
12. Japan Meteorological Agency (Ed.), Weather Business Now 2005. Oita: Saeki Printing, 2005 (in Japanese)
13. Kahneman, D., Tversky, A.: Prospect theory: An nnalysis of decision under risk. Econometrica 47(2), 263–291 (1979)
14. Katz, R.W., Murphy, A.H. (eds.): Economic Value of Weather and Climate Forecasts. Cambridge University Press, Cambridge (1997)
15. Murphy, A.H.: Decision making and the value of forecasts in a generalized model of the cost-loss ratio situation. Monthly Weather Review 113, 362–369 (1985)
16. Thompson, J.C.: A note on meteorological decision making. Journal of Applied Meteorology 5, 532–533 (1966)
17. Yager, R.R., Detyniecki, M., Bouchon-Meunier, B.: A context-dependent method for ordering fuzzy numbers using probabilities. IS 138(1–4), 237–255 (2001)
18. Wilks, D.S., Murphy, A.H.: The value of seasonal precipitation forecasts in a haying/pasturing problem in Western Oregon. Monthly Weather Review 113, 1738–1745 (1985)

Group Decision Making: From Consistency to Consensus

F. Chiclana[1], F. Mata[2], S. Alonso[3], E. Herrera-Viedma[3], and L. Martínez[2]

[1] Centre for Computational Intelligence, De Montfort University
Leicester LE1 9BH – UK
chiclana@dmu.ac.uk
[2] Department of Computer Science, University of Jaén
23700 Jaén – Spain
{fmata,martin}@ujaen.es
[3] Dept. of Computer Science and Artificial Intelligence, University of Granada
18071 Granada – Spain
{salonso,viedma}@decsai.ugr.es

Abstract. In group decision making (GDM) processes, prior to the selection of the best alternative(s), it would be desirable that experts achieve a high degree of consensus or agreement between them. Due to the complexity of most decision making problems, individuals' preferences may not satisfy formal properties. 'Consistency' is one of such properties, and it is associated with the *transitivity property*. Obviously, when carrying out a rational decision making, consistent information, i.e. information which does not imply any kind of contradiction, is more appropriate than information containing some contradictions. Therefore, in a GDM process, consistency should also be sought after.

In this paper we present a consensus model for GDM problems that proceeds from consistency to consensus. This model includes a novel consistency reaching module based on consistency measures. In particular, the model generates advice on how experts should change their preferences in order to reach a solution with high consistency and consensus degrees.

1 Introduction

Any decision making problem includes a selection process which involves, as part of it, the choice between the various alternatives solutions to the problem [14]. In GDM problems, however, it may happen that some experts from the group would not accept the group choice if they consider that their opinions have not been take into account 'properly'. Indeed, group choice should be based on the desires or preferences of 'all' the individuals in the group, a premise on which democratic theory is based on [3].

Preference relations are usually assumed to model experts' preferences in group decision making problems [4,12]. Classically, given two alternatives, an expert judges them in one of the following ways:

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 80–91, 2007.

(i) one alternative is preferred to another;
(ii) the two alternatives are indifferent to him/her;
(iii) he/she is unable to compare them.

However, given three alternatives x_i, x_j, x_k such that x_i is preferred to x_j and x_j to x_k, the question whether the "degree or strength of preference" of x_i over x_j exceeds, equals, or is less than the "degree or strength of preference" of x_j over x_k cannot be answered by the classical preference modelling. The implementation of the degree of preference between alternatives may be essential in many situations, and this can be modelled using fuzzy preference relations [1,2].

The main advantage of pairwise comparison is that of focusing exclusively on two alternatives at a time which facilitates experts when expressing their preferences. However, this way of providing preferences limits experts in their global perception of the alternatives and, as a consequence, the provided preferences could be not rational. Usually, rationality is related to consistency, which is associated with the *transitivity property* [8]. Many properties have been suggested to model transitivity of a fuzzy preference relation and, consequently, consistency may be measured according to which of these different properties is required to be satisfied. One of these properties is the "additive transitivity", which, as shown in [8], can be seen as the parallel concept of Saaty's consistency property in the case of multiplicative preference relations.

In any 'rational' decision making process, consistent information, i.e. information which does not imply any kind of contradiction, is more relevant or important than information containing some contradictions. As a consequence, in GDM processes consistency should also be sought after in order to make rational choices. To do this, experts should know how consistent they are. By letting the experts know their associated consistency measures at any moment, they could judge whether or not it is high enough. Also, with this information, expert would be able to analyse their preferences and make the necessary changes to their most inconsistent preference values to increase their global consistency.

In GDM situations, consensus between experts is usually searched using the basic rationality principles that each expert presents. Thus, consistency criteria should be first applied to fix the rationality of each expert and only afterwards experts' agreement should be obtained. If we were to secure consensus and only thereafter consistency, we could destroy the consensus in favour of the individual consistency and the final solution might not be acceptable for the group of expert.

In [5,6] a consensus model was proposed for GDM problems which used two types of measurements to guide the consensus reaching process [10]: *consensus degrees* to evaluate the agreement of all the experts, and *proximity degrees* to evaluate the distance between the experts' individual preferences and the group or collective ones. In [9] a consensus model which uses a recommendation module to help experts to change their preferences was presented. In [11] a consensus model with an adaptive recommendation module to the current level of agreement in each one of the consensus round was defined. In this paper, we continue improving that consensus model by incorporating a consistency criteria, and, when necessary, to advice experts on how to become more consistent. We define a new adaptive

consensus model in which once the experts provide their individual preference relations, consistency measures for each one are computed. These consistency measures are used to generate a consistency feedback mechanism that generates advice to the most inconsistent experts on the necessary changes to their most inconsistent preference values to increase their global consistency. Once the agreed minimum level of consistency has been reached, consensus is sought after.

The rest of the paper is set out as follows. Section 2 presents the structure of the new adaptive consensus model with its consistency control module. Section 3 describes in detail the consistency control module. Finally, Section 4 draws our conclusions.

2 An Adaptive Consensus Model with Consistency Control

In this section we present the structure of a new adaptive consensus model with consistency control. The structure of this new consensus model is depicted in Fig. 1.

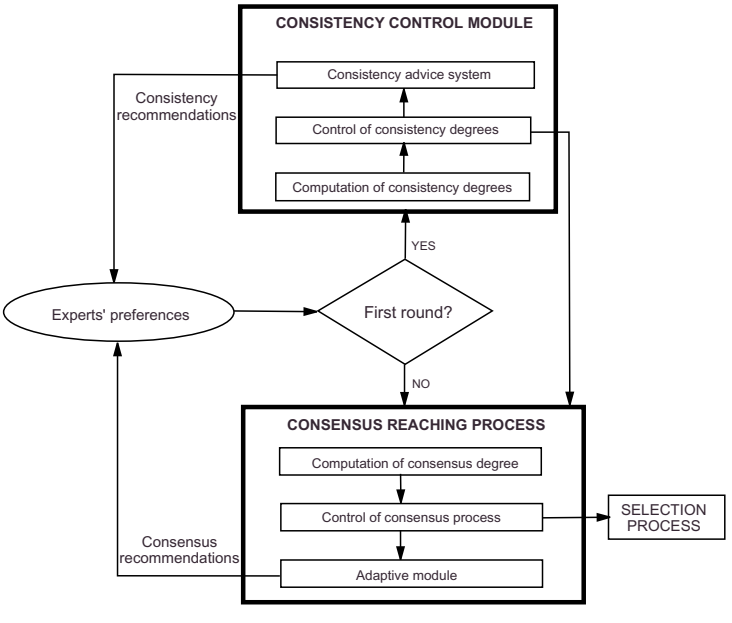


Fig. 1. Adaptive consensus model with consistency control

It is composed of two processes:

i) **Consistency Control Process.** Once experts' preferences are given, their consistency degrees are computed. If an expert is not consistent enough,

that expert will receive appropriate changes of preference values in order to increase his/her global consistency to an acceptable/agreed threshold level one. This process is explained in detail in Section 3.

ii) **Adaptive Consensus Reaching Process.** The consensus process is considered adaptive because the search for consensus is adapted to the current level of agreement among experts.

This adaptive process to achieve consensus among the group of experts consists of three steps:

1 Computation of consensus degrees. In this step consensus measures are computed for each fuzzy preference relation. In each consensus round, these measures are used to compute the level of agreement (consensus) between the experts of the group at the three different levels of a preference relation: pairs of alternatives, alternatives and preference relation.

2 Consensus control. In this step, it is decided whether to stop or to continue the application of the consensus process. This decision is based on the achievement or not of a fixed a priori consensus threshold value, $\gamma \in [0, 1]$, representing the minimum level of global agreement the experts should reach in order to proceed with the selection of the solution alternative to the problem.

3 Adaptive module. The consensus reaching process involves a procedure which identifies those preference values experts should change to achieve the desired agreement level. This identification is not fixed but it adapts to the current level (low, medium, high) of consensus computed in step 1:

3.1 Low consensus. Naturally, at the very beginning of the consensus process experts' preferences may differ substantially. In these cases the level of agreement could be quite low and a large number of experts' preferences should change in order to make the opinions closer. At this stage of the consensus process, and while the consensus is considered as 'low', 'all' experts are advised to change 'all' the preference values in which disagreement has been identified.

3.2 Medium consensus. In the 'intermediate' rounds of the consensus reaching process the consensus degree might not be considered as low anymore. In this stage of a consensus process, and while the consensus degree is considered as 'medium', only those experts furthest from the group as a collective will be advised to make changes on the preference values of those alternatives in which disagreement has been identified.

3.3 High consensus. When the level of consensus is approaching the consensus threshold value, only those experts furthest from the group as a collective will be advised to make changes on the preference values in which disagreement has been identified.

For more details on the described adaptive consensus reaching process the reader is referred to [11]. We should point out that the consistency control process is applied only in the first round of the consensus reaching process, because, as we shall show in the following section, when all the individual preference relations have associated a consistency degree above a particular minimum threshold

value then any weighted average collective preference relation will also have associated a consistency degree above that threshold value. Adding to this the fact that the consensus process tends to make the individual opinions closer to the collective ones [7], we conclude that individual consistency degrees will tend towards the collective one and therefore above the threshold value. Therefore, it is unnecessary to control the consistency level of each expert in each consensus round.

3 Consistency Control Module

The purpose of the consistency control module is to measure the level of consistency of each individual preference relation (expert) in order to identify the experts, alternatives and preference values most inconsistent within the GDM problem. This inconsistency identification is also used to suggest new 'consistent' preference values. The consistency control module develops its activity by means of three processes as illustrated in Fig. 2, which will be covered in the following subsections.

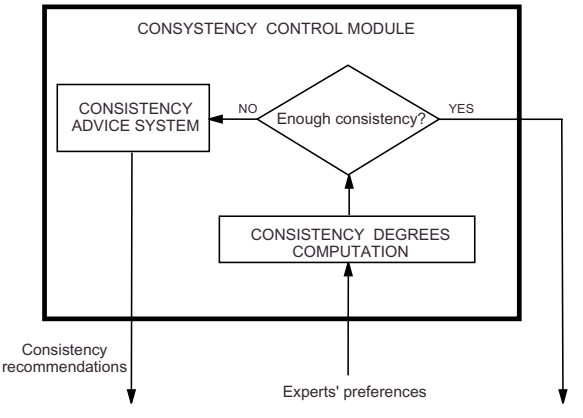


Fig. 2. Consistency control module

3.1 Computation of Consistency Degrees

In GDM problems with fuzzy preference relations some properties about the preferences expressed by the experts are usually assumed and desirable in order to avoid contradictions in their opinions, i.e, inconsistent opinions. One of these properties is associated with the transitivity in the pairwise comparison among any three alternatives. For fuzzy preference relations, transitivity has been modelled in many different ways due to the role the intensities of preference have (see [8]). In this paper, we make use of the *additive transitivity property.*

Being $P = (p_{ij})$ a fuzzy preference relation, the mathematical formulation of the *additive transitivity* was given by Tanino in [15]:

$$(p_{ij} - 0.5) + (p_{jk} - 0.5) = (p_{ik} - 0.5) \quad \forall i, j, k \in \{1, \ldots, n\} \tag{1}$$

As shown in [8,13] additive transitivity can be used to obtain more consistent fuzzy preference relation from a given one. Additive transitivity implies additive reciprocity. Indeed, because $p_{ii} = 0.5\ \forall i$, if we make $k = i$ in (1) then we have: $p_{ij} + p_{ji} = 1 \quad \forall i, j \in \{1, \ldots, n\}$. Then, (1) can be rewritten as:

$$p_{ik} = p_{ij} + p_{jk} - 0.5 \quad \forall i, j, k \in \{1, \ldots, n\} \tag{2}$$

We will consider a fuzzy preference relation P to be "*additive consistent*" when for every three alternatives in the problem $x_i, x_j, x_k \in X$ their associated preference degrees p_{ij}, p_{jk}, p_{ik} fulfil (2). An additive consistent fuzzy preference relation will be referred as consistent throughout the paper, as this is the only transitivity property we are considering.

Given a reciprocal fuzzy preference relation, (2) can be used to calculate an estimated value of a preference degree using other preference degrees. Indeed, using an intermediate alternative x_j, the following estimated value of p_{ik} $(i \neq k)$ is obtained:

$$ep_{ik}^{j} = p_{ij} + p_{jk} - 0.5 \tag{3}$$

The overall estimated value ep_{ik} of p_{ik} is obtained as the average of all possible values ep_{ik}^{j}, i.e.,

$$ep_{ik} = \sum_{\substack{j=1 \\ j \neq i,k}}^{n} \frac{ep_{ik}^{j}}{n-2}. \tag{4}$$

The value $|ep_{ik} - p_{ik}|$ can be used as a measure of the error between a preference value and its estimated one.

It is easy to obtain the expression of the estimated value of an estimated value, $e^2 p_{ik}$, which is:

$$e^2 p_{ik} = ep_{ik} + \frac{2}{n-2} \cdot (p_{ik} - ep_{ik})$$

This expression implies that the process of estimating preference values converges toward perfect consistency, which is expressed in the following proposition:

Proposition 1. *Let P be a reciprocal fuzzy preference relation. The following holds:*

$$\left| e^r p_{ik} - e^{r-1} p_{ik} \right| = \left(\frac{2}{n-2} \right)^{r-1} |ep_{ik} - p_{ik}|, \ r > 1.$$

When the information provided is completely consistent then $ep_{ik}^{j} = p_{ik}\ \forall j$. However, because experts are not always fully consistent, the information given by an expert may not verify (2) and some of the estimated preference degree

values ep_{ik}^{j} may not belong to the unit interval $[0,1]$. We note, from (3), that the maximum value of any of the preference degrees ep_{ik}^{j} is 1.5 while the minimum one is -0.5. In order to normalize the expression domains in the decision model the final estimated value of p_{ik} $(i \neq k)$, cp_{ik}, is defined as the median of the values 0, 1 and ep_{ik}:

$$cp_{ik} = med\{0, 1, ep_{ik}\}. \tag{5}$$

The error in $[0,1]$ between a preference value p_{ik} and its final estimated cp_{ik} is:

$$\varepsilon p_{ik} = |cp_{ik} - p_{ik}|. \tag{6}$$

Given a preference value $p_{ik} \in [0,1]$, the following holds $|ep_{ik} - p_{ik}| = |ep_{ik} - cp_{ik}| + |cp_{ik} - p_{ik}|$ and therefore $\varepsilon p_{ik} \leq |ep_{ik} - p_{ik}|$ $\forall i, k$.

For being $P = (p_{ij})$ reciprocal, it is obvious that the preference relation $CP = (cp_{ik})$ is also reciprocal and $\varepsilon p_{ik} = \varepsilon p_{ki}$. We interpret $\varepsilon p_{ik} = 0$ as a situation of total consistency between p_{ik} (p_{ki}) and the rest of information in P. Obviously, the higher the value of εp_{ik} the more inconsistent is p_{ik} (p_{ki}) with respect to the rest of information in P.

This interpretation allows us to evaluate the consistency in each one of the three different levels of a reciprocal fuzzy preference relation P:

Level 1. *Consistency degree associated to a pair of alternatives* p_{ik} (p_{ki}),

$$cd_{ik} = 1 - \varepsilon p_{ik} \tag{7}$$

Level 2. *Consistency degree associated to an alternative* x_i,

$$cd_i = \sum_{\substack{k=1 \\ k \neq i}}^{n} \frac{cd_{ik}}{n-1} \tag{8}$$

When $cd_i = 1$ all the preference values involving the alternative x_i are fully consistent, otherwise, the lower cd_i the more inconsistent these preference values are with respect to the rest of information in P.

Level 3. *Consistency degree of the reciprocal fuzzy preference relation*,

$$cd = \sum_{i=1}^{n} \frac{cd_i}{n} \tag{9}$$

When $cd = 1$ the reciprocal fuzzy preference relation P is fully consistent, otherwise, the lower cd the more inconsistent P.

The computation of the estimated values and consistency degrees for a reciprocal preference relation are illustrated in the following example:

Example 1. *The following are a reciprocal fuzzy preference relation and its reciprocal estimated fuzzy preference relation*

$$P = \begin{pmatrix} - & 0.7 & 0.9 & 0.5 \\ 0.3 & - & 0.6 & 0.7 \\ 0.1 & 0.4 & - & 0.8 \\ 0.5 & 0.3 & 0.2 & - \end{pmatrix} \longrightarrow CP = \begin{pmatrix} - & 0.55 & 0.5 & 1.0 \\ 0.45 & - & 0.55 & 0.6 \\ 0.5 & 0.45 & - & 0.35 \\ 0.0 & 0.4 & 0.65 & - \end{pmatrix}$$

The value $cp_{14} = 1$ *has been obtained as follows:*

$$ep_{14} = \frac{ep_{14}^2 + ep_{14}^3}{2} = \frac{0.9 + 1.2}{2} = 1.05 \Rightarrow cp_{14} = med\{0, 1, 1.05\} = 1.$$

The consistency degrees at the three levels of the preference relation are:

Level 1. *Consistency degrees at the level of pairs of alternatives*

$$CD = \begin{pmatrix} - & 0.85 & 0.6 & 0.5 \\ 0.85 & - & 0.95 & 0.9 \\ 0.6 & 0.95 & - & 0.55 \\ 0.5 & 0.9 & 0.55 & - \end{pmatrix}$$

Level 2. *Consistency degree of each alternative:*

$$cd_1 = 0.65 \quad cd_2 = 0.9 \quad cd_3 = 0.7 \quad cd_4 = 0.65$$

Level 3. *Consistency degree of the relation:*

$$cd = 0.73.$$

Let $P^c = (p_{ij}^c)$ be a weighted mean collective preference relation obtained from a set of reciprocal fuzzy preference relations $\{P^1, \ldots, P^m\}$. The estimated value of the collective preference value $p_{ij}^c = \sum_{l=1}^n w_l \cdot p_{ij}^l$, with $w_l \geq 0 \ \forall l$ and $\sum_l w_l = 1$, is $ep_{ij}^c = \sum_{\substack{k=1 \\ j \neq i,k}}^n \frac{(ep_{ij}^c)^k}{n-2}$, with $(ep_{ij}^c)^k = p_{ik}^c + p_{kj}^c - 0.5$. Putting these expressions together we get:

$$ep_{ij}^c = \sum_{\substack{k=1 \\ j \neq i,k}}^n \frac{\sum_{l=1}^n w_l \cdot p_{ik}^l + \sum_{l=1}^n w_l \cdot p_{kj}^l - 0.5}{n-2} = \sum_{\substack{k=1 \\ j \neq i,k}}^n \frac{\sum_{l=1}^m w_l \cdot (p_{ik}^l + p_{kj}^l - 0.5)}{n-2}$$
$$= \sum_{l=1}^m w_l \frac{\sum_{\substack{k=1 \\ j \neq i,k}}^n p_{ik}^l + p_{kj}^l - 0.5}{n-2} = \sum_{l=1}^m w_l \cdot ep_{ij}^l$$

This result is summarised in the following proposition:

Proposition 2. *The estimated collective preference relation of a weighted mean collective preference obtained from a set of reciprocal fuzzy preference relations is also the weighted mean of the estimated individual preference relations.*

The error between a collective preference value and its estimated one is

$$\left| ep_{ij}^c - p_{ij}^c \right| = \left| \sum_{l=1}^m w_l \cdot ep_{ij}^l - \sum_{l=1}^n w_l \cdot p_{ij}^l \right| = \left| \sum_{l=1}^m w_l \cdot (ep_{ij}^l - p_{ij}^l) \right|$$

Using the well known properties $|a + b| \leq |a| + |b|$ and $|a \cdot b| = |a| \cdot |b|$ we have

$$\left| \sum_{l=1}^m w_l \cdot (ep_{ij}^l - p_{ij}^l) \right| \leq \sum_{l=1}^m w_l \cdot \left| ep_{ij}^l - p_{ij}^l \right| \leq \max_l \left| ep_{ij}^l - p_{ij}^l \right|$$

Therefore, we have proved that

Proposition 3. *The error between a collective preference value and its estimated one is lower or equal to the maximum error between the individual preference values and their estimated ones.*

From $\varepsilon p_{ij} \leq |ep_{ij} - p_{ij}|$ and proposition 3 we can easily prove that when all estimated values of each one of the individual preference relations of the set $\{P^1, \ldots, P^m\}$ are in $[0, 1]$ the consistency degree of a weighted mean collective preference relation will be greater or equal than the minimum of the individual consistency degrees, $\min_l cd^l$. When one or more individual estimated values are not in $[0, 1]$ then this limit is reduced by the quantity $\sum_{i,j;j\neq i} \frac{\max_l |ep_{ij}^l - cp_{ij}^l|}{n(n-1)}$. In a situation of high individual consistency degrees the distance between $cp_{ij}^l = med\{0, 1, ep_{ik}^l\}$ and p_{ij}^l will be small (or zero) and as a consequence the distance between ep_{ij}^l and cp_{ij}^l will also be small (or zero). All this together can be used to claim that in GDM problems in which all experts provide highly consistent preferences the (weighted mean) collective preference will also be highly consistent.

3.2 Consistency Control

We assume that before providing any preferences the group of experts agree on a threshold consistency degree value (β) for an expert to be considered as consistent. After providing preferences, experts' associated consistency degrees are obtained, $cd^i \forall i$. If all experts are consistent, i.e. $cd^i \geq \beta \ \forall i$, then the consensus reaching process is applied. Otherwise, a consistency advice system is applied i) to identify the inconsistent experts, alternatives, and preference values; and ii) to generate an alternative consistent value for each one of the inconsistent preference values.

3.3 Consistency Advice System

This system suggests experts some changes on the most inconsistent preference values. To do so, the following three steps are carried out:

1. To identify those experts (l) in the group with a global consistency level (cd^l) lower than the minimum threshold consistency value (β).
2. To identify for each one of these experts those alternatives (i) with a consistency degree (cd_i^l) lower than β.
3. To identify for each one of these alternatives the preference values whose consistency level (cd_{ij}^l) is lower than β.
   The set of preference values to be recommended for change will be:

$$\{(l, i, j) | \max\{cd^l, cd_i^l, cd_{ij}^l\} < \beta\}.$$

Based on proposition 1, a preference value of the above set (p_{ij}^l) will be recommended to be changed to a value closer to its final estimated value (cp_{ij}^l). This change will bring the original individual preference relation (P^l) closer to

its estimated one (CP^l) and therefore it will become more consistent globally. Thus, if $cd^l_{ij} < \beta$, in order to reach the minimum threshold value, p^h_{ij} will be recommended to be changed to

$$\bar{p}^l_{ij} = p^l_{ij} + sign(cp^l_{ij} - p^l_{ij}) \cdot (\beta - cd^l_{ij}),$$

where $sign(X)$ returns the sign of X. Finally, in order to maintain reciprocity, the value p^l_{ji} will be recommended to be changed to $\bar{p}^l_{ji} = 1 - \bar{p}^l_{ij}$.

Example 2. *Suppose we have a set of four experts providing the following fuzzy preference relations on a set of four alternatives:*

$$P^1 = \begin{pmatrix} - & 0.2 & 0.6 & 0.4 \\ 0.8 & - & 0.9 & 0.7 \\ 0.4 & 0.1 & - & 0.3 \\ 0.6 & 0.3 & 0.7 & - \end{pmatrix} \quad P^2 = \begin{pmatrix} - & 0.7 & 0.9 & 0.5 \\ 0.3 & - & 0.6 & 0.7 \\ 0.1 & 0.4 & - & 0.8 \\ 0.5 & 0.3 & 0.2 & - \end{pmatrix}$$

$$P^3 = \begin{pmatrix} - & 0.3 & 0.5 & 0.7 \\ 0.7 & - & 0.1 & 0.3 \\ 0.5 & 0.9 & - & 0.25 \\ 0.3 & 0.7 & 0.75 & - \end{pmatrix} \quad P^4 = \begin{pmatrix} - & 0.25 & 0.15 & 0.65 \\ 0.75 & - & 0.6 & 0.8 \\ 0.85 & 0.4 & - & 0.5 \\ 0.35 & 0.2 & 0.5 & - \end{pmatrix}$$

Let the threshold value be $\beta = 0.8$. We have the following global consistency values:

$$cd^1 = 1,\ cd^2 = 0.73,\ cd^3 = 0.63;\ cd^4 = 0.82.$$

This means that recommendations of change will be given to experts e_2, e_3. For these two experts we have the following consistency degree matrices and consistency degree of alternatives:

$$CD^2 = \begin{pmatrix} - & 0.85 & 0.6 & 0.5 \\ 0.85 & - & 0.95 & 0.9 \\ 0.6 & 0.95 & - & 0.55 \\ 0.5 & 0.9 & 0.55 & - \end{pmatrix};\quad cd^2_1 = 0.65,\ cd^2_2 = 0.9,\ cd^2_3 = 0.7,\ cd^2_4 = 0.65$$

$$CD^3 = \begin{pmatrix} - & 0.4 & 0.93 & 0.48 \\ 0.4 & - & 0.48 & 0.93 \\ 0.93 & 0.48 & - & 0.55 \\ 0.48 & 0.93 & 0.55 & - \end{pmatrix};\quad cd^3_1 = 0.6,\ cd^3_2 = 0.6,\ cd^3_3 = 0.65,\ cd^3_4 = 0.65$$

The recommended new preference values would be:

e_2**:** $\bar{p}_{13} = 0.7$ ($\bar{p}_{31} = 0.3$); $\bar{p}_{14} = 0.8$ ($\bar{p}_{41} = 0.2$); $\bar{p}_{34} = 0.55$ ($\bar{p}_{43} = 0.45$).
e_3**:** $\bar{p}_{12} = 0.7$ ($\bar{p}_{21} = 0.3$); $\bar{p}_{14} = 0.38$ ($\bar{p}_{41} = 0.62$); $\bar{p}_{23} = 0.43$ ($\bar{p}_{32} = 0.57$); $\bar{p}_{34} = 0.5$ ($\bar{p}_{43} = 0.5$).

If these recommended values were assumed by these experts, their new global consistency values would become $cd^2 = 0.94$ and $cd^3 = 0.92$, which represent a considerable improvement regarding their previous global consistency levels. Afterwards the consensus and selection process are carried out.

4 Conclusions

In any rational GDM process, both consensus and consistency should be sought after. In this paper we have addressed the issues of measuring consistency and of achieving a high level of consistency within an adaptive consensus reaching process. For doing that, we have developed a consistency advice module, based on theoretical results, for recommending 'consistent' changes to experts for the most inconsistent preference values. We have argued that consistency is needed to be checked just once before the application of the consensus process, because (a) when all individual experts provide highly consistent preferences the (weighted mean) collective preference will also be highly consistent, and (b) the consensus process tends to make the individual opinions closer to the collective ones. Also, if we were to secure consensus and only thereafter consistency, we could destroy the consensus in favour of the individual consistency and the final solution could not be acceptable for the group of expert.

Acknowledgements

This work was prepared during Dr Francisco Mata's research visit to the Centre for Computational Intelligence at De Montfort University (Leicester, UK) in the summer 2006. This work has been supported by the Research Projects TIN2006-02121 and SAINFOWEB-PAI00602.

References

1. Chiclana, F., Herrera, F., Herrera-Viedma, E.: Integrating three representation models in fuzzy multipurpose decision making based on fuzzy preference relations. Fuzzy Sets and Systems 97(1), 33–48 (1998)
2. Chiclana, F., Herrera, F., Herrera-Viedma, E.: A Note on the Internal Consistency of Various Preference Representations. Fuzzy Sets and Systems 131(1), 75–78 (2002)
3. Fishburn, P.C.: The Theory of Social Choice. Princeton University Press, Princeton, NJ (1973)
4. Fishburn, P.C.: Utility Theory for Decision Making. Robert E. Krieger Publishing Company (1979)
5. Herrera, F., Herrera-Viedma, E., Verdegay, J.L.: A model of consensus in group decision making under linguistic assessments. Fuzzy Sets and Systems 79, 73–87 (1996)
6. Herrera, F., Herrera-Viedma, E., Verdegay, J.L.: Linguistic measures based on fuzzy coincidence for reaching consensus in group decision making. International Journal approximate Reasoning 16, 309–334 (1997)
7. Herrera-Viedma, E., Herrera, F., Chiclana, F.: A consensus model for multiperson decision making with different preference structures. IEEE Transactions on Systems, Man and Cybernetics-Part A: Systems and Humans 32(3), 394–402 (2002)
8. Herrera-Viedma, E., Herrera, F., Chiclana, F., Luque, M.: Some issues on consistency of fuzzy preference relations. European Journal of Operational Research 154(1), 98–109 (2004)

9. Herrera-Viedma, E., Martínez, L., Mata, F., Chiclana, F.: A consensus support system model for group decision-making problems with multi-granular linguistic preference relations. IEEE Transactions on Fuzzy Systems 13(5), 644–658 (2005)
10. Herrera-Viedma, E., Mata, F., Martínez, L., Chiclana, F., Pérez, L.G.: Measurements of consensus in multi-granular linguistic group decision making. In: Torra, V., Narukawa, Y. (eds.) MDAI 2004. LNCS (LNAI), vol. 3131, pp. 194–204. Springer, Heidelberg (2004)
11. Herrera-Viedma, E., Mata, F., Martínez, L., Pérez, L.G.: An adaptive module for the consensus reaching process in group decision making problems. In: Torra, V., Narukawa, Y., Miyamoto, S. (eds.) MDAI 2005. LNCS (LNAI), vol. 3558, pp. 89–98. Springer, Heidelberg (2005)
12. Kacprzyk, J.: Group decision making with a fuzzy linguistic majority. Fuzzy Sets and Systems 18, 105–118 (1986)
13. Ma, J., Fan, Z.-P., Jiang, Y.-P., Mao, J.Y., Ma, L.: A method for repairing the inconsistency of fuzzy preference relations. Fuzzy Sets and Systems 157, 20–33 (2006)
14. Roubens, M.: Fuzzy sets and decision analysis. Fuzzy Sets and Systems 90, 199–206 (1997)
15. Tanino, T.: Fuzzy preference orderings in group decision making. Fuzzy Sets and Systems 12, 117–131 (1984)

Weighting Individual Opinions in Group Decision Making

José Luis García-Lapresta

Dep. de Economía Aplicada, Universidad de Valladolid, PRESAD Research Group
Avda. Valle de Esgueva 6, 47011 Valladolid, Spain
lapresta@eco.uva.es
http://www2.eco.uva.es/lapresta

Abstract. In this paper we introduce a multi-stage decision making procedure where decision makers sort the alternatives by means of a fixed set of linguistic categories, each one has associated a numerical score. First we average the scores obtained by each alternative and we consider the associated collective preference. Then, we obtain a distance between each individual preference and the collective one through the Euclidean distance among the individual and collective scoring vectors. Taking into account these distances, we measure the agreement in each subset of decision makers, and a weight is assigned to each decision maker: his/her overall contribution to the agreement. Those decision makers whose overall contribution to the agreement are not positive are expelled and we re-initiate the decision procedure with only the opinions of the decision makers which positively contribute to the agreement. The sequential process is repeated until it determines a final subset of decision makers where all of them positively contribute to the agreement. Then, we apply a weighted procedure where the scores each decision maker indirectly assigns to the alternatives are multiplied by the weight of the corresponding decision maker, and we obtain the final ranking of the alternatives.

1 Introduction

When a group of decision makers have to decide a collective ranking of a set of alternatives, usually they rank the alternatives and then an aggregation procedure is applied for generating the collective order. If the number of alternatives is high, then decision makers can have difficulties in the task of ranking feasible alternatives. According to Dummett [7]: "If there are, say, twenty possible outcomes, the task of deciding the precise order of preference in which he ranks them may induce a kind of psychological paralysis in the voter".

In order to facilitate decision makers to arrange the alternatives, we propose that decision makers sort the alternatives within a small set of linguistic categories (for instance, *excellent*, *very good*, *good*, *regular*, *bad* and *very bad*).[1]

[1] The use of linguistic information within the decision making framework has been widely used in the literature. See, for instance, Yager [12] and Herrera and Herrera-Viedma [10].

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 92–103, 2007.

We assign a score to each linguistic category and then a collective score is associated with each alternative by means of the average of the individual scores. Consequently, the alternatives are ordered by the obtained collective scores.

After this first stage, we introduce a distance among individual opinions and the aggregated weak order. Through these distances, we propose an index for measuring the overall contribution to the agreement of each decision maker.[2] Those decision makers whose indices are not positive will be excluded, and we re-initiate the process with only the opinions of the individuals which positively contribute to the agreement. We repeat this procedure, by recalculating the overall indices, until obtaining a final subset of decision makers where all of them positively contribute to the agreement. Then, we weight the scores that decision makers (indirectly) assign to the alternatives by their overall contribution to the agreement indices, and we obtain the final collective ranking of the alternatives.

Notice that weighting individual opinions with the mentioned indices, decision makers are incentivated to not declare very divergent opinions with respect to the mean opinion. Otherwise, they can be penalized by reducing their influence over the collective ranking or being eliminated of the group.

The paper is organized as follows. Section 2 is devoted to introduce the notation and the main notions needed in the multi-stage decision procedure, which we present in Section 3. Finally, Section 4 includes some concluding remarks.

2 Preliminaries

Let $V = \{v_1, \ldots, v_m\}$ a set of decision makers (or voters) who show their preferences on the pairs of a set of alternatives $X = \{x_1, \ldots, x_n\}$, with $m, n \geq 3$. $\mathcal{P}(V)$ denotes the power set of V ($I \in \mathcal{P}(V) \Leftrightarrow I \subseteq V$). Linear orders are binary relations satisfying reflexivity, antisymmetry and transitivity, and weak orders (or complete preorders) are complete and transitive binary relations. With $|I|$ we denote the cardinal of I.

We consider that each decision maker classifies the alternatives within a set of *linguistic categories* $\mathcal{L} = \{l_1, \ldots, l_p\}$, with $p \geq 2$, linearly ordered $l_1 > \cdots > l_p$. The *individual assignment* of the decision maker v_i is a mapping $C_i : X \longrightarrow \mathcal{L}$ which assigns a linguistic category $C_i(x_u) \in \mathcal{L}$ to each alternative $x_u \in X$.

Associated with C_i, we consider the weak order R_i defined by $x_u \, R_i \, x_v$ if $C_i(x_u) \geq C_i(x_v)$. With P_i and I_i we denote, respectively, the asymmetric (strict preference) and symmetric (indifference) relations associated with R_i, i.e., $x_u \, P_i \, x_v$ whenever not $x_v \, R_i \, x_u$, and $x_u \, I_i \, x_v$ whenever $x_u \, R_i \, x_v$ and $x_v \, R_i \, x_u$.

It is important to note that decision makers are not totally free in declaring their preferences. They have to adjust their opinions to the set of linguistic categories, so the associated weak orders depend on the way they sort the alternatives within the fixed scheme provided by $\mathcal{L}$. Even more, given a weak order

[2] The use of metrics for aggregating individual preferences has been analyzed in the literature by many authors (see, for instance, Kemeny [11], Cook and Seiford [5,6], Armstrong, Cook and Seiford [1] and Cook, Kress and Seiford [4]).

R_i with no more than p equivalence classes, it is possible to define different individual assignments. For instance, given the weak order $x_1\, I_i\, x_2\, P_i\, x_3\, P_i\, x_4\, I_i\, x_5$, for $p = 4$ we can associate the assignment: $C_i(x_1) = C_i(x_2) = l_1$, $C_i(x_3) = l_2$ and $C_i(x_4) = C_i(x_5) = l_4$; but also $C_i(x_1) = C_i(x_2) = l_1$, $C_i(x_3) = l_2$ and $C_i(x_4) = C_i(x_5) = l_3$; and so on.

A *profile* is a vector $\mathbf{C} = (C_1, \ldots, C_m)$ of individual assignments. We denote by $\mathcal{C}$ the set of profiles.

We assume that every linguistic category $l_k \in \mathcal{L}$ has associated a score $s_k \in \mathbb{R}$ in such a way that $s_1 \geq s_2 \geq \cdots \geq s_p$ and $s_1 > s_p = 0$. For the decision maker v_i, let $S_i : X \longrightarrow \mathbb{R}$ be the mapping which assigns the score to each alternative, $S_i(x_u) = s_k$ whenever $C_i(x_u) = l_k$. The *scoring vector of* v_i is $(S_i(x_1), \ldots, S_i(x_n))$.

Naturally, if $s_i > s_j$ for all $i, j \in \{1, \ldots, p\}$ such that $i > j$, then each linguistic category is univoquely determined by its associated score. Thus, given the scoring vector of a decision maker we directly know the way this individual sort the alternatives. Although linguistic categories are equivalent to decreasing sequences of scores, there exist clear differences from a behavioral point of view.

Example 1. Consider three decision makers who sort the alternatives of $X = \{x_1, , \ldots, x_9\}$ according to the set of linguistic categories $\mathcal{L} = \{l_1, \ldots, l_6\}$ and the associated scores given in Table 1.

Table 1. Linguistic categories

$\mathcal{L}$	Meaning	Score
l_1	Excellent	$s_1 = 8$
l_2	Very good	$s_1 = 5$
l_3	Good	$s_1 = 3$
l_4	Regular	$s_1 = 2$
l_5	Bad	$s_1 = 1$
l_6	Very bad	$s_1 = 0$

In Table 2 we include the way of decision makers sort the alternatives within the set of linguistic categories.

It seems reasonable to assign as collective score $S(x_u)$, for each alternative $x_u \in X$, the average of the individual scores:

$$S(x_u) = \frac{1}{m} \sum_{i=1}^{m} S_i(x_u).$$

Taking into account the *average collective scoring vector*, $(S(x_1), \ldots, S(x_n))$, we define the *average collective weak order* on X:

$$x_u\, R\, x_v \;\Leftrightarrow\; S(x_u) \geq S(x_v).$$

Table 2. Sorting alternatives

	R_1	R_2	R_3
l_1	x_1	x_3	
l_2	$x_3\ x_6$	x_8	$x_1\ x_2\ x_5$
l_3	$x_2\ x_4\ x_5\ x_8$	$x_4\ x_6$	$x_4\ x_6$
l_4	x_7	$x_1\ x_9$	x_3
l_5	x_9	x_5	$x_7\ x_8$
l_6		$x_2\ x_7$	x_9

Table 3. Scores

	S_1	S_2	S_3	S
x_1	8	2	5	5
x_2	3	0	5	2.666
x_3	5	8	2	5
x_4	3	3	3	3
x_5	3	1	5	3
x_6	5	3	3	3.666
x_7	2	0	1	1
x_8	3	5	1	3
x_9	1	2	0	1

Following Example 1, in Table 3 we show the individual and collective scores obtained by each alternative.

In Table 4 we show the collective preference provided by the weak order R.

Table 4. Collective order

$$\begin{array}{ccccc} & & x_1 & x_3 & \\ & & x_6 & & \\ x_4 & & x_5 & & x_8 \\ & & x_2 & & \\ & & x_7 & x_9 & \end{array}$$

If we compare the collective preference with the individual ones in Example 1, it is clear that there exist some differences. In order to have some information about the agreement in each subset of decision makers, we firstly introduce a distance between pairs of preferences (scoring vectors). Since the arithmetic mean minimizes the sum of distances to individual values with respect to the Euclidean metric, it seems reasonable to use this metric for measuring the distance among scoring vectors.

Definition 1. *Let* $(S(x_1),\dots,S(x_n)),(S'(x_1),\dots,S'(x_n))$ *be two individual or collective scoring vectors. We define the distance between these vectors by means of the Euclidean metric:*

$$d(S,S')=\sqrt{\sum_{u=1}^{n}(S(x_u)-S'(x_u))^2}.$$

Taking into account Example 1, the distances among the individual opinions and the collective preference are given by:

$$d(S_1,S)=3.448<d(S_3,S)=4.887<d(S_2,S)=5.962\,. \tag{1}$$

In next section we introduce an index which measures the overall contribution to the agreement for each decision maker. By means of these measures, we modify the initial group decision procedure for priorizating consensus[3].

3 The Multi-stage Decision Making Procedure

In order to introduce our multi-stage group decision making procedure, we first consider a specific agreement measure which is based on the distances among individual and collective scoring vectors in each subset of decision makers.

We note that Bosch [2] introduced a general concept of *consensus measure* within the class of linear orders by assuming three axioms: Unanimity, Anonymity (symmetry with respect to decision makers) and Neutrality (symmetry with respect to alternatives).

Definition 2. *The mapping* $\mathcal{M}:\mathcal{C}\times\mathcal{P}(V)\longrightarrow[0,1]$ *defined by*

$$\mathcal{M}(\mathbf{C},I)=\begin{cases}1-\dfrac{\sum\limits_{i\in I}d(S_i,S)}{|I|\,s_1\sqrt{n}}, & \text{if } I\neq\emptyset,\\[2ex] 0, & \text{if } I=\emptyset\end{cases}$$

is called overall agreement measure.

We note that $s_1\sqrt{n}$ is the maximum distance among scoring vectors, clearly between $(S(x_1),\dots,S(x_n))=(s_1,\dots,s_1)$ and $(S'(x_1),\dots,S'(x_n))=(0,\dots,0)$:

$$d(S,S')=\sqrt{n\,s_1^2}=s_1\sqrt{n}.$$

Then, $\mathcal{M}(\boldsymbol{C},I)\in[0,1]$, for every $(\boldsymbol{C},I)\in\mathcal{C}\times\mathcal{P}(V)$.

[3] Along the paper we do not talk about consensus, but about agreement. The reason is that consensus has different meanings. One of them is related to an interactive and sequential procedure where decision makers have to change their preferences in order to improve the agreement. Usually, a moderator advise decision makers to modify some opinions (see, for instance, Eklund, Rusinowska and de Swart [8]).

It is important to note that $\mathcal{M}(\boldsymbol{C}, I) = 1$ if and only if $d(S_i, S) = 0$ for every $v_i \in I$, i.e., $S_i = S_j$ and $C_i = C_j$ for all $v_i, v_j \in I$; in other words, $\mathcal{M}(\boldsymbol{C}, I) = 1$ if and only if all the decision makers share the same assignment (Unanimity).

The problem of determine the minimum agreement (or total disagreement) presents more difficulties, because in the case of more than 2 decision makers agreement and disagreement are not symmetric notions (see Bosch [2]).

It is easy to see that our overall agreement measure satisfies the other axioms of Bosch [2], Anonymity and Neutrality.

We now introduce an index which measures the overall contribution to the agreement of each voter with respect to a fixed profile, by adding up the marginal contributions to the agreement in all the subsets of decision makers.

Definition 3. *The* overall contribution to the agreement *of decision maker* v_i *with respect to a profile* $\mathbf{C} \in \mathcal{C}$ *is defined by:*

$$w_i = \sum_{I \subseteq V} \Big(\mathcal{M}(\mathbf{C}, I) - \mathcal{M}(\mathbf{C}, I \setminus \{v_i\}) \Big).$$

Obviously, if $v_i \notin I$, then $\mathcal{M}(\boldsymbol{C}, I) - \mathcal{M}(\boldsymbol{C}, I \setminus \{v_i\}) = 0$. If $w_i > 0$, we say that decision maker v_i positively contributes to the agreement; and if $w_i < 0$, we say that decision maker v_i negatively contributes to the agreement.

We now introduce a new collective preference by weighting the scores which decision makers (indirectly) assign to alternatives with the corresponding overall contribution to the agreement indices.

Definition 4. *The* collective weak order *associated with the weighting vector* $\boldsymbol{w} = (w_1, \ldots, w_m)$, $R^{\boldsymbol{w}}$, *is defined by*

$$x_u \, R^{\boldsymbol{w}} \, x_v \;\Leftrightarrow\; S^{\boldsymbol{w}}(x_u) \geq S^{\boldsymbol{w}}(x_v),$$

where

$$S^{\boldsymbol{w}}(x_u) = \frac{1}{m} \sum_{i=1}^{m} w_i \cdot S_i(x_u).$$

Consequently, we prioritize the decision makers in order of their contribution to the agreement (see Cook, Kress and Seiford [4]).

Notice that the average collective weak order is just the collective weak order associated with the weighting vector $\boldsymbol{w} = (1, \ldots, 1)$.

Example 2. Following Example 1 and the overall contributions to the agreement introduced in Definition 3, we obtain $w_1 = 0.670$, $w_2 = 0.557$ and $w_3 = 0.605$. If we apply these weights in the collective decision procedure of Definition 4, then the opinion of the first decision maker counts $w_1/w_2 = 1.203$ times the opinion of the second one; $w_1/w_3 = 1.107$ times the opinion of the second one; and the opinion of the third decision maker counts $w_3/w_2 = 1.087$ times the opinion of the second one.

In Table 5 we show the initial collective scores given in Table 3 and the new collective scores after we weight the opinions of the decision makers with the overall contributions to the agreement. We also include the ratio between the new collective scores, $S^{\boldsymbol{w}}$, and the initial collective scores, S. These differences are due to the individual contributions to the agreement. It is important to note that in the new version of the decision procedure there are not ties.

Table 5. New collective scores

	S	$S^{\boldsymbol{w}}$	$S^{\boldsymbol{w}}/S$
x_1	5	3.168	0.633
x_2	2.666	1.679	0.629
x_3	5	3.006	0.601
x_4	3	1.833	0.611
x_5	3	1.865	0.621
x_6	3.666	2.280	0.621
x_7	1	0.648	0.648
x_8	3	1.800	0.600
x_9	1	0.594	0.594

According to the obtained weights, the new version of the decision procedure linearly order the alternatives, by means of $R^{\boldsymbol{w}}$, in the following way:

$$x_1,\ x_3,\ x_6,\ x_5,\ x_4,\ x_8,\ x_2,\ x_7,\ x_9. \tag{2}$$

Since negative values of w_i could artificially alter the outcomes of $R^{\boldsymbol{w}}$, we consider the weighting vector $\boldsymbol{w}' = (w'_1, \ldots, w'_m)$, where $w'_i = \max\{w_i, 0\}$. This problem is now analyzed through an example.

3.1 An Illustrative Example

Consider four decision makers who sort the alternatives of $X = \{x_1, , \ldots, x_9\}$ according to the set of linguistic categories $\mathcal{L} = \{l_1, \ldots, l_6\}$ and the associated scores given in Table 1. Table 6 contains the way these decision makers rank the alternatives. In Table 7 we show the individual and collective scores obtained by each alternative, and Table 8 includes the collective preference provided by the weak order R.

The overall contributions to the agreement are $w_1 = 0.387$, $w_2 = 0.151$, $w_3 = -0.204$ and $w_4 = 0.197$. According to these indices, the weighted decision procedure (Definition 4) linearly order the alternatives in the following way:

$$x_4,\ x_5,\ x_3,\ x_1,\ x_2,\ x_7,\ x_8,\ x_6,\ x_9. \tag{3}$$

Since the third decision maker negatively contributes to the agreement, then his/her associated scores are multiplied by a negative weight. In order to avoid this undesirable effect, we will consider non negative weights $w'_i = \max\{w_i, 0\}$:

Table 6. Sorting alternatives

	R_1	R_2	R_3	R_4
l_1	x_3	x_1	$x_2\ x_6\ x_9$	x_5
l_2	$x_1\ x_2\ x_4$	x_4	$x_3\ x_7\ x_8$	x_4
l_3	x_5	x_5	x_1	$x_2\ x_3\ x_7\ x_8$
l_4	$x_6\ x_7$	$x_7\ x_8$		x_9
l_5	$x_8\ x_9$	$x_2\ x_3\ x_6$	x_4	x_1
l_6		x_9	x_5	x_6

Table 7. Scores

	S_1	S_2	S_3	S_4	S
x_1	5	8	3	1	4.25
x_2	5	1	8	3	4.25
x_3	8	1	5	3	4.25
x_4	5	5	1	5	4
x_5	3	3	0	8	3.5
x_6	2	1	8	0	2.75
x_7	2	2	5	3	3
x_8	1	2	5	3	2.75
x_9	1	0	8	2	2.75

$w'_1 = 0.387,\ w'_2 = 0.151,\ w'_3 = 0$ and $w'_4 = 0.197$. Applying again the decision procedure, we obtain a new linear order on the set of alternatives:

$$x_3,\ x_4,\ x_1,\ x_5,\ x_2,\ x_7,\ x_8,\ x_6,\ x_9. \tag{4}$$

Note that x_3 is ranked in the third position in (3) and it is the first alternative in (4). Since in (3), $S_3(x_3) = 5$ has been multiplied by the negative weight $w_3 = -0.204$, this alternative has been penalized. However, in (4) the opinion of the third decision maker has not been considered. This fact joint with the first decision maker, with the highest weight $w_1 = 0.387$, ranks x_3 at the first alternative, induce that this alternative reaches the top position.

Although the new ranking (4) is more appropriate than (3) for reflecting the individual opinions, it is important to note that all the calculations have been made taking into account the opinions of the third decision maker. If we think that the third decision maker judgments should not be considered (because his/her divergent opinions with respect to the global opinion), we can start a new step of the decision procedure where only the opinions of the rest of the decision makers are taken into account.

In Table 10 we show the individual and collective scores obtained by each alternative, and Table 11 contains the collective preference provided by the weak order R.

Table 8. Collective order

$$\begin{array}{ccc} x_1 & x_2 & x_3 \\ & x_4 & \\ & x_5 & \\ & x_7 & \\ & x_6 & \\ x_8 & & x_9 \end{array}$$

Table 9. Sorting alternatives

	R_1	R_2	R_3	R_4
l_1	x_3	x_1	$x_2\ x_6\ x_9$	x_5
l_2	$x_1\ x_2\ x_4$	x_4	$x_3\ x_7\ x_8$	x_4
l_3	x_5	x_5	x_1	$x_2\ x_3\ x_7\ x_8$
l_4	$x_6\ x_7$	$x_7\ x_8$		x_9
l_5	$x_8\ x_9$	$x_2\ x_3\ x_6$	x_4	x_1
l_6		x_9	x_5	x_6

The new overall contributions to the agreement are

$$w_1^{(2)} = 0.583 > w_2^{(2)} = 0.570 > w_4^{(2)} = 0.566,$$

while

$$w_1^{(1)} = w_1 = 0.387 > w_4^{(1)} = w_4 = 0.197 > w_2^{(1)} = w_2 = 0.151.$$

These differences are due to the fact that in the second iteration of the decision procedure the divergent opinions of the third decision maker have not been considered.

According to the weights $w_1^{(2)}$, $w_2^{(2)}$, $w_4^{(2)}$, the new stage of the decision procedure linearly order the alternatives in the following way:

$$x_4,\ x_1,\ x_5,\ x_3,\ x_2,\ x_7,\ x_8,\ x_6,\ x_9. \tag{5}$$

Clearly, there exist important differences among the linear orders provided by (3), (4) and (5). In fact, (3) takes into account the divergent opinions of the third decision maker; (4) does not consider the opinions of the third decision maker, but the collective ranking and, consequently, all the weights are based on the opinions of all the decision makers, including that of the divergent third decision maker; finally, (5) totally excludes the opinions of the third decision maker.

3.2 Scheme of the Multi-stage Group Decision Procedure

In 3.1 we have analyzed through examples how aggregate individual opinions by considering the overall contributions to the agreement. We now present the considered multi-stage decision procedure in a general and precise way.

Table 10. Scores

	S_1	S_2	S_4	S
x_1	5	8	1	4.666
x_2	5	1	3	3
x_3	8	1	3	4
x_4	5	5	5	5
x_5	3	3	8	4.666
x_6	2	1	0	1
x_7	2	2	3	2.333
x_8	1	2	3	2
x_9	1	0	2	1

Table 11. Collective order

$$\begin{array}{ccc} & x_4 & \\ x_1 & & x_5 \\ & x_3 & \\ & x_2 & \\ & x_7 & \\ & x_8 & \\ x_6 & & x_9 \end{array}$$

1. Decision makers $V = \{v_1, \ldots, v_m\}$ sort the alternatives of $X = \{x_1, \ldots, x_n\}$ according to the linguistic categories of $\mathcal{L} = \{l_1, \ldots, l_p\}$. Then, we obtain individual weak orders $R_1, \ldots, R_m$ which rank the alternatives within the fixed set of linguistic categories.
2. Taking into account the scores $s_1, \ldots, s_p$ associated with $l_1, \ldots, l_p$, a score is assigned to each alternative for every decision maker: $S_i(x_u),\ i = 1, \ldots, m,$ $u = 1, \ldots, n$.
3. We aggregate the individual opinions by means of collective scores which are defined as the average of the individual scores:

$$S(x_u) = \frac{1}{m} \sum_{i=1}^{m} S_i(x_u)$$

and we rank the alternatives through the collective weak order R:

$$x_u \, R \, x_v \ \Leftrightarrow \ S(x_u) \geq S(x_v).$$

4. We calculate the overall contributions to the agreement (Definition 3) for all the decision makers: $w_1, \ldots, w_m$.
   (a) If $w_i \geq 0$ for every $i \in \{1, \ldots, m\}$, then we obtain the new collective scores by:

$$S^{\boldsymbol{w}}(x_u) = \frac{1}{m} \sum_{i=1}^{m} w_i \cdot S_i(x_u)$$

and we rank the alternatives by means of the collective weak order $R^{\boldsymbol{w}}$:

$$x_u \, R^{\boldsymbol{w}} \, x_v \;\Leftrightarrow\; S^{\boldsymbol{w}}(x_u) \geq S^{\boldsymbol{w}}(x_v).$$

(b) Otherwise, we eliminate those decision makers whose overall contributions to the agreement are not positive. We now initiate the decision procedure for the remaining decision makers $V^+ = \{v_i \in V \mid w_i > 0\}$.

4 Concluding Remarks

Usually decision makers have difficulties to rank order a high number of alternatives. In order to facilitate this task, we have considered a mechanism where decision makers sort alternatives through a small set of linguistic categories. We associate a score to each linguistic category and then we aggregate the individual opinions by means of the average of the individual scores, providing a collective weak order on the set of alternatives. Then we assign an index to each decision maker which measures his/her overall contribution to the agreement. Taking into account these indices, we weight individual scores and we obtain a new collective ranking of alternatives after excluding the opinions of those decision makers whose overall contributions to the agreement are not positive. The new collective ranking of alternatives provides the final decision.

Since overall contribution to the agreement indices (which multiply individual scores) usually are irrational numbers, it is unlikely that the weighted procedure provides ties among alternatives.

Since the proposed decision procedure penalizes those individuals that are far from consensus positions, this fact incentives decision makers to moderate their opinions. Otherwise, they can be excluded or their opinions can be underestimated. However, it is worth emphasizing that our proposal only requires a single judgement to each individual about the alternatives.

We can generalize our group decision procedure by considering different aggregation operators (see Fodor and Roubens [9] and Calvo, Kolesárova, Komorníková and Mesiar [3]) for obtaining the collective scores. Another way of generalization consists in measuring distances among individual and collective scoring vectors by means of different metrics.

Acknowledgments

This research started during a stay of the author in the University of Tilburg. The author thanks Harrie de Swart for his hospitality, and Rob Bosch for facilitating us his Ph. Dissertation. The funding support of the Spanish Ministerio de Educación y Ciencia (Project SEJ2006-04267/ECON), ERDF and Junta de Castilla y León (Consejería de Educación y Cultura, Project VA040A05) are gratefully acknowledged.

References

1. Armstrong, R.D., Cook, W.D., Seiford, L.M.: Priority ranking and concensus formation: The case of ties. Management Science 8, 638–645 (1982)
2. Bosch, R.: Characterizations of Voging Rules and Consensus Measures. Ph. D. Dissertation, Tilburg University (2005)
3. Calvo, T., Kolesárova, A., Komorníková, M., Mesiar, R.: Aggregation operators: Properties, classes and constructions models. In: Calvo, T., Mayor, G., Mesiar, R. R. (eds.) Aggregation Operators: New Trends and Applications, pp. 3–104. Physica-Verlag, Heidelberg (2002)
4. Cook, W.D., Kress, M., Seiford, L.M.: A general framework for distance-based consensus in ordinal ranking models. European Journal of Operational Research 96, 392–397 (1996)
5. Cook, W.D., Seiford, L.M.: Priority ranking and concensus formation. Management Science 24, 1721–1732 (1978)
6. Cook, W.D., Seiford, L.M.: On the Borda-Kendall consensus method for priority ranking problems. Management Science 28, 621–637 (1982)
7. Dummett, M.: Voting Procedures. Clarendon Press, Oxford (1984)
8. Eklund, P., Rusinowska, A., de Swart, H.: Consensus reaching in committees. European Journal of Operational Research 178, 185–193 (2007)
9. Fodor, J., Roubens, M.: Fuzzy Preference Modelling and Multicriteria Decision Support. Kluwer Academic Publishers, Dordrecht (1994)
10. Herrera, F., Herrera-Viedma, E.: Linguistic decision analysis: Steps for solving decision problems under linguistic information. Fuzzy Sets and Systems 115, 67–82 (2000)
11. Kemeny, J.: Mathematics without numbers. Daedalus 88, 571–591 (1959)
12. Yager, R.R.: Non-numeric multi-criteria multi-person decision making. Group Decision and Negotiation 2, 81–93 (1993)

An Active Learning Method Based on Most Possible Misclassification Sampling Using Committee

Jun Long, Jianping Yin, and En Zhu

School of Computer Science, National University of Defense Technology,
Changsha 410073, P.R. China
jdragon_nudt@hotmail.com

Abstract. By selecting and asking the user to label only the most informative instances, active learners can significantly reduce the number of labeled training instances to learn a classification function. We focus here on how to select the most informative instances for labeling. In this paper we make three contributions. First, in contrast to the leading sampling strategy of halving the volume of version space, we present the sampling strategy of reducing the volume of version space by more than half with the assumption of target function being chosen from nonuniform distribution over version space. Second, via Halving model, we propose the idea of sampling the instances that would be most possibly misclassified. Third, we present a sampling method named CBMPMS (Committee Based Most Possible Misclassification Sampling) which samples the instances that have the largest probability to be misclassified by the current classifier. Comparing the proposed CBMPMS method with the existing active learning methods, when the classifiers achieve the same accuracy, the former method will sample fewer times than the latter ones. The experiments show that the proposed method outperforms the traditional sampling methods on most selected datasets.

1 Introduction

Supervised learning methods construct classifiers using labeled instances as the training set. However, labeling instances may be time-consuming, tedious, error prone and costly in some tasks (e.g., document categorization and gene expression analysis). To cope with this problem, active learning algorithms [1] were proposed which select the most informative instances and ask the user to label them. Thus the burden of labeling large number of instances could be alleviated.

There are two main types of active learning methods according to the source of unlabeled instances: pool-based and stream-based. The former assumes that the active learner has access to a pool of unlabeled instances; when labeling, the active learner could go through the entire pool and select the most informative one or ones. In contrast, in the stream-based scenario, successive instances from a stream of unlabeled set are presented to the active learner which should decide whether or not label the current instance. In this paper, we only focus on the pool-based scenario.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 104–113, 2007.

In general, the pool-based active learning method comprises two parts: a learning engine and a sampling engine [2]. The whole process of active learning could be described as follows. Initially, the system has a training set of labeled instances and a set of unlabeled instances. Then, the learning engine trains a classifier on the original training set. After that, the sampling engine chooses the most informative instances from the unlabeled instances and requests a human expert to label it before it is added into the training set. Then the learning engine constructs a new classifier on the updated training set. The whole process runs repeatedly until the evaluation index of the classifier or iteration times reaches the preset value.

Depending on the criterion used to select instances to label, the current research fall under several categories: uncertainty reduction, expected-error minimization and version space reduction. The uncertainty reduction [1] approach selects the instances on which the current classifier has the least certainty to predict. Many sampling methods apply the similar strategy [4, 5, 6]. They perform better than random sampling in most tasks, but sometimes they may select outliers. The expected-error minimization approach [7, 8] samples the instances that minimize the future expected error rate on the test set. Such methods expect to achieve the lowest error, but they are computationally expensive. The version space reduction approach tries to select the instances that can reduce the volume of version space by half. Query-by-Committee [9, 10] is a representative method of this approach that constructs a committee consists of randomly selected hypotheses from the version space and selects the instances on which the disagreement within the committee is the greatest. The version space reduction approach also includes sg-net [11], QBag [12], QBoost [12] and Active Decorate [13].

In this paper, we make three contributions. First, in contrast to the leading sampling strategy of halving the volume of version space, we present the sampling strategy of reducing the volume of version space by more than half with the assumption of target function being chosen from nonuniform distribution over version space. Second, we illustrate the idea of sampling the instances that would be most possibly misclassified via Halving model. Third, we present a sampling method named CBMPMS (Committee Based Most Possible Misclassification Sampling) which samples the instances that have the largest probability to be misclassified by the current classifier. Comparing the proposed CBMPMS method with the existing active learning methods, when the classifiers achieve the same accuracy, the former method will sample fewer times than the latter ones.

In general, our CBMPMS method has several characters. First, a measurement of misclassification is used as the sampling criterion. Second, multi-class problems could also be handled by our CBMPMS method. Third, instead of class label, our sampling criterion focuses on the class probability distribution given by the base classifier which is more precise.

The following sections of the paper are organized as follows. Section 2 describes the general process of active learning and how sampling influences the version space. Section 3 describes the proposed active learning method CBMPMS. Section 4 shows the experimental results of the CBMPMS method as well as other methods on many data sets. Section 5 draws the conclusion.

2 Background on Active Learning

2.1 General Process of Active Learning

The active learning framework can be denoted by $< L, UL, I, S, M, ES >$, where

L denotes the set of labeled instances;

UL denotes the set of unlabeled instances;

I denotes the classifier;

S denotes the stopping criterion which could be the accuracy to reach or the iteration times;

M denotes the number of instances which would be sampled in each iteration;

ES denotes the effectiveness score。

The general process of active learning framework is as follows:

Input:

An initial labeled set L, an unlabeled set UL, a classifier I, a stopping criterion S, and an integer M that specify the number of instances sampled in each iteration.

1. Train the classifier I by L;
2. While stopping criterion S is not satisfied, execute (1)-(6)
   - (1) For each example $x_i \in UL$ compute ES_i, the effectiveness;
   - (2) Select a subset A of size M from UL based on ES_i;
   - (3) Remove A from UL;
   - (4) Label instances in A;
   - (5) Add A into L;
   - (6) Train the classifier I by L.

Output:

The classifier I trained by the final labeled set L.

2.2 The Influence of Sampling on Version Space

In active learning problem, we denote the target function to learn by $\theta_0(X)$, and the parametric learning model by $\theta(W, X)$. Both $\theta_0(X)$ and $\theta(W, X)$ are Boolean-valued functions. X is an independent variable and W is a vector of model parameters. So learning is to find out a vector w_0 such that $\theta_0(X) = \theta(w_0, X)$ for all X. Then the version space W_t after the t th instance was sampled will be:

$$W_t = \{w \mid \theta(w, x_i) = \theta_0(x_i), i = 1, ..., t\} \tag{1}$$

Then, the reduction of version space after the $(t+1)$ th instance was sampled will be:

$$\Delta W_t = W_t - W_{t+1} = \{w \mid \theta(w, x_{t+1}) \neq \theta_0(x_{t+1}), \theta(w, x_i) = \theta_0(x_i), i = 1, ..., t\} \quad (2)$$

Obviously, the bigger $\#(\Delta W_t)$ is, the faster the volume of version space will be reduced and the more informative the $(t+1)$ th instance will be.

Generally, for binary classification, it is thought that the best sampling method is selecting the instance which could make about half of hypotheses in the version space predict as positive and half of hypotheses predict as negative. Thus about half of hypotheses in the version space could be deleted at each sampling time. Then the target function could be found with about $\log_2 | H |$ sampling. So, the leading sampling methods (i.e. uncertainty method) are based on this idea.

3 CBMPMS Method

3.1 Reducing the Version Space Size by More Than Half

Actually, the target function is chosen from some distribution π over version space. The strategy of halving the size of version space is based on an assumption that the distribution π uniform over version space. Under this assumption, the probability of a hypothesis being the target function is:

$$p(w_i = w_o \mid D) = \frac{1}{t}, i = 1, \cdots, t \quad (3)$$

where w_i is a hypothesis in the version space, w_o is the target function, t is the number of the hypotheses in the version space and D is the training set. In this situation, halving the size of version space is the best way to find the target function as quickly as possible.

However, the distribution π may not be uniform over version space. Then, the probability of a hypothesis being the target function would be:

$$p(w_i = w_o \mid D) = \alpha_i, i = 1, \cdots, t \quad (4)$$

where $\sum_{i=1}^{t} \alpha_i = 1$ and each α_i may not equal to the same value. Then, the strategy of halving the size of version space is not appropriate for this situation. To address such problem, we present an intuitive sampling strategy which is aiming to reducing the version space size by more than half. As each α_i may not be equal to the same value, it is possible to pick a few hypotheses with the largest α_i and keep them in the version space when sampling. Thus, the volume of version space would be reduced by more than half when sampling. Actually, it is so hard to know the exact value of α_i that this strategy can not be used directly. But, this strategy yields the idea of sampling the most possibly misclassified instances which is explained via halving model in the following section.

3.2 Most Possible Misclassification Sampling

Halving model is a machine learning model that maintains a description of the version space. It gives each hypothesis in the version space equal right to vote. At the training stage, the hypotheses whose predictions agree with the instance's true label would be kept unremoved, while the hypotheses whose predictions do not agree with the instance's true label would be eliminated. When the classifier meets an instance to predict, all hypotheses in the version space vote for its label and the resulting label will be the one that gets a majority vote.

Consider the following idea: after the halving model predicts the new instance's label, if we could know the true label of it, we can check whether the instance is misclassified. When the predicted label is the same as the true label, it means the majority of hypotheses in version space predict right and the remainder hypotheses which predict wrong should be eliminated. When the predicted label is not the same as the true label, it means the majority of hypotheses in version space predict wrong and should be eliminated.

So, if we can sample the misclassified instance, halving model would eliminate the majority of hypotheses in version space. Thus it would converge faster than those methods halving the version space size. The CBMPMS method is based on such an idea. It is quite different from the idea of the uncertainty method. The CBMPMS method aims to select the instance which the classifier believes to belong to one category but actually not. In contrast, the uncertainty sampling method selects the instance which almost has the equal probability to belong to either category judged by the current classifier.

To measure the degree of an instance x_i being misclassified, the parameter R_i is defined as:

$$R_i = L(P(Y \mid x_i), P_D(Y \mid x_i)) \tag{5}$$

where L is a loss function that measures the differences between its two input variables, Y is the variable which an instance could be classified as and $P(Y \mid x)$ is a vector of which each element is the probability of an instance x being classified as each possible value of Y. Then $P(Y \mid x_i)$ denotes the true class probability distribution (CPD) and $P_D(Y \mid x_i)$ denotes the CPD which is got by the classifier trained on the training set D. Thus, R_i measures the difference between the true CPD and the CPD got by the classifier trained on the training set.

The loss function L has a variety of candidates, and after careful comparison and testing, the chosen one is:

$$L(P(Y \mid x_i), P_D(Y \mid x_i)) = -\sum_{y \in Y} (\chi_i \log \chi_i) \tag{6}$$

where χ_i denotes $\left|P(y \mid x_i) - P_D(y \mid x_i)\right|$. Then R_i is the entropy of the absolute difference between $P(Y \mid x_i)$ and $P_D(Y \mid x_i)$. It is picked for several reasons: First, it is

theoretically well founded on the information theory; second, its counterparts perform not well in the test.

Furthermore, the true class probability distribution $P(Y \mid x_i)$ is unknown, so we should estimate it.

As Hansen and Salamon [14] have pointed out, a committee can be more accurate than its component classifiers when each component classifier outputs independently and has an accuracy over 1/2. Moreover, they also introduce that the more component classifiers used, the less likelihood of an error by a majority decision rule. Furthermore, the committee error rate goes to zero with an infinite size of the committee [14]. Hence, it is reasonable to estimate the true CPD using committee on the assumption that each component classifier responses independently and has an accuracy over 1/2. As a result, the estimate of R_i is:

$$R_i = -\sum_{y \in Y} (\left| P_D^{com}(y \mid x_i) - P_D(y \mid x_i) \right|) \log(\left| P_D^{com}(y \mid x_i) - P_D(y \mid x_i) \right|) \tag{7}$$

where $P_D^{com}(y \mid x_i)$ is the class probability distribution of x_i predicted by the committee consisting of random selected hypotheses in the version space on the training set D.

3.3 The Process of the CBMPMS Method

The process of CBMPMS method is given as follows:

Input:

an initial labeled set L, an unlabeled set UL, a classifier I, a stopping criterion S, and an integer M which specify the number of instances sampled in each iteration.

1. Train the classifier I by L;
2. While stopping criterion S not met, execute (1)-(6)

(1) For each example $x_i \in UL$ compute

$$ES_i = -\sum_{y \in Y} (\left| P_D^{com}(y \mid x_i) - P_D(y \mid x_i) \right|) \log(\left| P_D^{com}(y \mid x_i) - P_D(y \mid x_i) \right|);$$

(2) Select a subset A of size M from UL in which instances have the biggest ES_i;

(3) Remove A from UL;

(4) Label instances in A;

(5) Add A into L;

(6) Train the classifier I by L.

Output:

The classifier I trained by the final labeled set L.

---

4 Experiments

Experiments were conducted to compare the performance of our CBMPMS method with other active learning methods. Five different active learning algorithms were tested:

- Random sampling: choosing the instance at random.
- Uncertainty sampling: choosing the instance with the largest label uncertainty, as in [1].
- QBC sampling: choosing the instance that the committee members disagree with most, as in [9].
- Error-Reduction sampling: choosing the instance that maximizes the reduction in the total predicted label entropy, as in [8].
- CBMPMS sampling (the method introduced in this paper): choosing the instance that most likely to be predicted wrong.

The experiments were done on 18 representative datasets from machine learning repository provided by UCI [15]. Naive bayes was selected to be the base classifier.

For lack of space we can not give the details of all the tests. So we displayed the results of only a few selected tests that were most representative in figures and summarized the data utilization of the different active learners in Table 1. We define data utilization as the number of instances an active learner requires to reach the target error rate. 10-fold cross-validation is used to compare the performance of the selected active learner on the 18 datasets. All results presented were averages of ten runs.

In Table 1, the least data utilization is marked in bold in each row and the number of wins is presented in the last row.

Table 1. Data utilization of the different active learners

	Random	*Uncertainty*	*QBC*	*Error-reduction*	*CBMPMS*	*Target Accuracy (%)*
car	148.9	**83.8**	134.0	672.8	91.6	80.11
ionosphere	43.8	68.1	**12.9**	46.3	29.2	86.04
mushroom	2598.1	38.1	**28.3**	1690.5	37.2	95.00
vote	172.3	74.8	107.2	87.7	**67.7**	95.01
waveform-5000	42.0	84.5	32.7	41.1	**27.9**	76.08
nursery	98.7	**71.9**	221.5	2796.9	118.8	84.74
anneal	254.7	271.4	243.8	354.2	**202.9**	84.28
balance-scale	51.6	283.6	53.7	**27.9**	143.7	80.89
colic	220.8	350.9	236.3	**69.7**	209.6	76.61
credit-g	157.2	393.2	211.6	295.8	**119.0**	75.02
cylinder-bands	**19.8**	123.5	44.9	42.5	36.8	63.65
kr-vs-kp	163.0	**48.7**	84.8	1396.5	144.4	84.22
mfeat-fourier	192.0	153.1	138.6	213.0	**130.6**	72.10
mfeat-pixel	154.9	**86.2**	157.6	139.6	87.2	88.97
segment	142.1	291.3	**92.0**	509.6	115.8	77.63
soybean	262.0	229.1	206.4	299.0	**111.9**	91.47
splice	136.7	**91.6**	118.4	414.7	170.0	88.21
vowel	227.8	300.0	169.0	434.0	**121.6**	61.66
No. of Wins	1	5	3	2	**7**	

According to Table 1, it shows that our CBMPMS sampling method has a superior performance than other sampling methods on most datasets. Moreover, our CBMPMS method performs better than the Uncertainty method on 13 datasets and the QBC method on 12 datasets. Based on these results, we can conclude that using the idea of CBMPMS is much likely to reduce the size of version space by more than half. For those datasets which the CBMPMS method is defeated, there may be some reasons. First, the estimated true class probability distribution given by the committee may not be precise when the committee has a small size for some special tasks or the component hypothesis has an accuracy less than 1/2. Second, there may be some better loss functions or estimated true CPD than the current one we choose on those datasets. In future work we will deal with these problems.

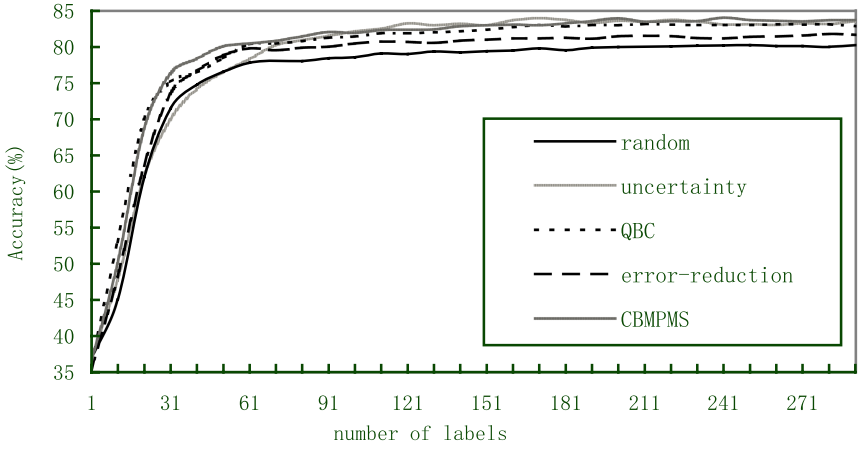


Fig. 1. Average testing accuracy on waveform-5000 dataset

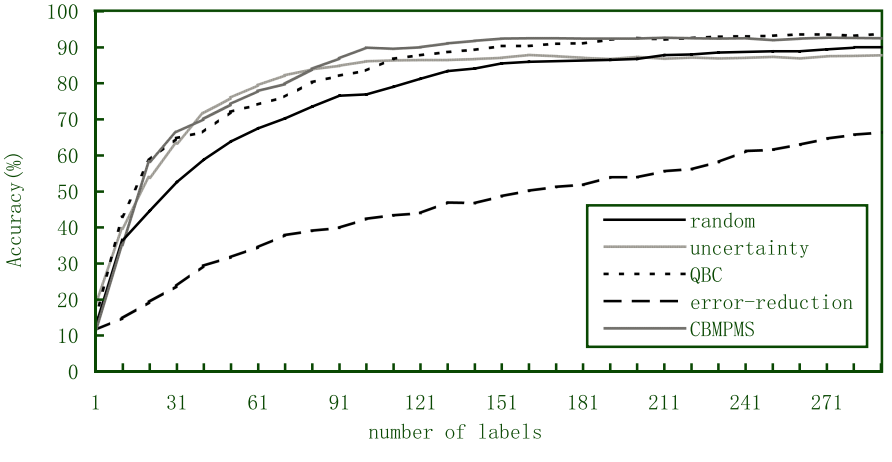


Fig. 2. Average testing accuracy on soybean dataset

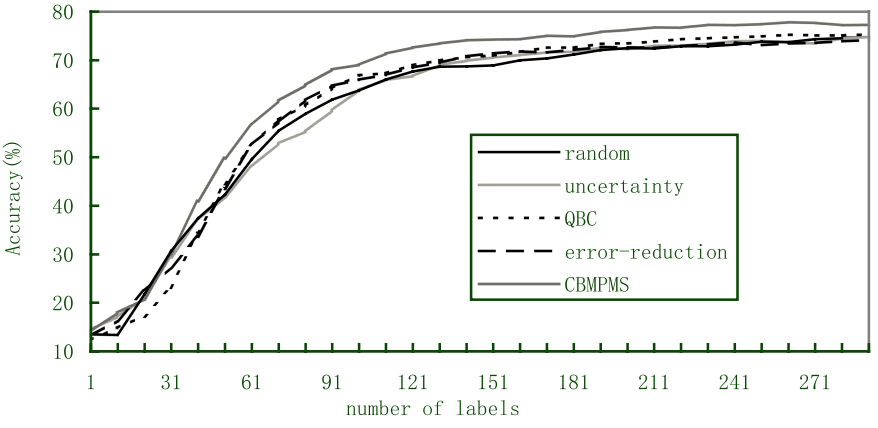


Fig. 3. Average testing accuracy on mfeat-fourier dataset

Figure 1, 2 and 3 show the results on the datasets of waveform-5000, soybean and mfeat-fourier, respectively. In all these figures, the vertical axis shows the accuracy of the classifier and the horizontal axis shows the number of labels.

In figure 1, it shows that our CBMPMS method outperforms the other methods. The random method achieves its maximal accuracy 80.22% at about the 291st sampling. Our CBMPMS method requires 61 sampling to obtain the same accuracy.

Figure 2 shows the accuracy of the CBMPMS method reaches 92.24% at the 151st sampling while the uncertainty method reaches the same accuracy at 291st sampling. Surprisingly, the error-reduction method does a little worse than the random method. There might be two reasons. First, the random method might be a very competitive method, which could get an expected generalization error decrease of $O(1/n^{\alpha})$ where n is the training set size and α depends on the task and learning method [3]. Second, the error-reduction method uses loss function to estimate the expected error of the classifier, which is not precise. So when the unlabeled instances provide some false expected error information, the error-reduction method would lose some efficiency.

In figure 3, our CBMPMS method outperforms the other active learners throughout the learning curve. The QBC method gets its highest accuracy, about 75.22%, after 291 sampling while the CBMPMS method reaches the same accuracy at the 191st sampling. Furthermore, the other methods even can not get the accuracy of 75.22%.

5 Conclusion

Focusing on the sampling question in pool-based active learning, we analyze the influence of sampling on version space and propose a strategy to reduce the volume of version space by more than half which is different from the current sampling methods that halve the size of version space. Based on this strategy, we proposed the idea of sampling the instances that would be most possibly misclassified. Then, we present

the CBMPMS method, which samples the instances with the highest probability of misclassification by the current classifier. It has an advantage on converging speed over other sampling methods and could be effective in active learning area. Experiments show that the CBMPMS method is efficient and practical.

We would like to pursue following directions: calculating more precise local accuracy of the classifier, providing the theoretical proof of the converging speed of the version space and application of the CBMPMS method in the real world area.

Acknowledgments. This research was supported by the National Natural Science Foundation of China (No. 60603015, 60603062).

References

1. Lewis, D.D., Gale, W.A.: A sequential algorithm for training text classifiers. In: Proc. of SIGIR-94. 17th ACM International Conference on Research and Development in Information Retrieval, pp. 3–12. Springer, Heidelberg (1994)
2. Hieu, T., Arnold, S.: Active Learning Using Pre-clustering. In: Proc. 21th International Conf. on Machine Learning, Banff, Canada (2004)
3. Schein, A.I.: Active learning for logistic regression: [Ph D dissertation] (2004)
4. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. Journal of Machine Learning Research 2, 45–66 (2001)
5. Schohn, G., Cohn, D.: Less is more: Active learning with support vector machines. In: Proc. 17th International Conf on Machine Learning, pp. 839–846. Morgan Kaufmann, San Francisco (2000)
6. Campbell, C., Cristianini, N., Smola, A.: Query learning with large margin classifiers. In: Proc. 17th International Conf. on Machine Learning, pp. 111–118. Morgan Kaufmann, San Francisco (2000)
7. Cohn, D.A., Ghahramani, Z., Jordan, M.I.: Active learning with statistical models. Journal of Artificial Intelligence research 4, 129–145 (1996)
8. Roy, N., McCallum, A.: Toward optimal active learning through sampling estimation of error reduction. In: Proc. 18th International Conf. on Machine Learning, pp. 441–448. Morgan Kaufmann, San Francisco, CA (2001)
9. Seung, H.S., Opper, M., Sompolinsky, H.: Query by committee. Computational Learning Theory, 287–294 (1992)
10. Freund, Y., Seung, H.S., Shamir, E., Tishby, N.: Selective sampling using the query by committee algorithm. Machine Learning 28, 133–168 (1997)
11. Cohn, D.A., Ladner, R.: Improving generalization with active learning. Machine Learning 15, 201–221 (1994)
12. Abe, N., Mamitsuka, H.: Query learning using boosting and bagging. In: Proc. 15th International Conf on Machine Learning, pp. 1–10. Morgan Kaufmann, Madison, CA (1998)
13. Melville, P., Mooney, R.J.: Diverse ensembles for active learning. In: Proc. 21th International Conf. on Machine Learning, pp. 584–591. Morgan Kaufmann, Banff, CA (2004)
14. Hansen, L., Salamon, P.: Neural network ensembles. IEEE Trans. Pattern Analysis and Machine Intell. 12, 993–1001 (1990)
15. Newman, D.J., Hettich, S., Blake, C.L., Merz, C.J.: UCI Repository of machine learning databases. Irvine, CA: University of California, Department of Information and Computer Science (1998), [http://www.ics.uci.edu/ mlearn/MLRepository.html]

Research on Next Generation Grids Using a Fuzzy Assessment Method

Chunyang Ye[1,2], Feng Xiong[1,2], and Qi Luo[3,4]

[1] School of Management, Wuhan University of Technology, Wuhan 430070, China
[2] Ningbo Universitiy, Ningbo, 315211, China
[3] College of Engineering and Technology, Southwest University, Chongqing 400715, China
[4] Information Engineering school,Wuhan University of Science and Technology and Zhongnan Branch,Wuhan 430223, China
YeChunYang@nbu.edu.cn

Abstract. Main issues to be faced by next-generation Grids are the management and exploitation of the overwhelming amount of data produced by applications but also by Grid operation, and the intelligent use of Grid resources and services. To achieve these very ambitious goals, next-generation Grids should include knowledge discovery and knowledge management functionalities, for both applications and system management. This paper proposes a fuzzy assessment method. The assessment of Next-generation Grids is composed of objective and subjective assessment. The results manifest that the assessment method is effective through testing in some system.

Keywords: Next-generation Grids, fuzzy assessment, knowledge discovery, knowledge management.

1 Introduction

Main issues to be faced by next-generation Grids are the management and exploitation of the overwhelming amount of data produced by applications but also by Grid operation, and the intelligent use of Grid resources and services [1]. To achieve these very ambitious goals, next-generation Grids should include knowledge discovery and knowledge management functionalities, for both applications and system management. The way how data and information available at different levels of Grid can be effectively acquired, represented, exchanged, integrated, and converted into useful knowledge is an emerging research field known as Grid Intelligence.

The solutions that will be developed will certainly driven by the previous needs and requirements, but will also leverage and probably integrate some key technologies and methodologies emerging in many computer science fields, apparently far and unaware of Grids, such as peer-to-peer and ubiquitous computing, ontology-based reasoning, and knowledge management[2][3]. In particular, ontologies and metadata are the basic elements through which Grid Intelligence services can be deployed. Using ontologies, Grids may offer semantic modeling of user's tasks, available services, and data sources to support high level services and dynamic services finding and composition.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 114–120, 2007.

Moreover, data mining and knowledge management techniques could enable high level services based on the semantics of stored data. Such services could be employed both at operation layer, where Grid management could gain from information hidden into data, and at application layer, where user could be able to exploit distributed data repository, using the Grid not only for high-performance access, movement and processing of data, but also to apply key analysis tools and instruments.

The assessment of next-generation Grids can help developers know the weaknesses of next-generation Grids so as to take effective improvement actions. It can also enable users to select a suitable next-generation Grids to solve problems. An ideal assessment method should consider social, cultural, economic and technical factors. In the proposed method, the objective and subjective assessment criteria are established by taking into account the experiences of the next- generation Grids developers, the software assessment standards, and the suggestions of users. The objective assessment criteria consist of two parts: the criteria commonly used in website assessment and those related to knowledge organization and management. We use the available network testing software to obtain the former and develop a system to test the latter. The subjective assessment about the quality of knowledge service carries out through the cooperation between experts and agents. The overall performance assessment of the next-generation Grids is determined by the membership functions of the integrated assessment value. Finally, we propose an optimum solution to improve the performance of next-generation Grids.

2 Next-Generation Grids Frame

To face the growing complexity of Grids and the overwhelming amount of data to be managed, main requirements of future Grids will be: knowledge discovery and knowledge management functionalities, system management, semantic modeling of user's tasks, Grid services, data sources, computing devices, to offer high level services and dynamic services finding and composition.

In particular, to fulfill some of the requirements listed before, we envision that next-generation Grids should first provide the following three main classes of services and related architectural framework:

Knowledge management and ontology-based services. They are used to build, manipulate, and interoperate, in a homogeneous way, the Grid knowledge base. The term Grid knowledge base is used here to indicate all the data stored, maintained, and updated by the Grid, both for user, application, and operation purposes. It comprises, for example, the Globus MDS data and metadata, the Grid services usage data, the application data sources and results, etc. Many of these data are currently maintained by Grid middleware or by Grid applications, so the very main challenge for next-generation Grids will be their seamless integration and utilization. From an architectural point of view, technologies useful for building, manipulating, and reasoning on the Grid knowledge base are ontologies and logic programming. In such scenario, each object on the Grid is classified through one or more ontologies into the knowledge base. Two important services that could be offered are: ontology-based Grid programming and request-resource matchmaking. In the next

Section we will show a simple example of component-based programming based on domain ontology.

Knowledge discovery services. They are used to extract knowledge from the data stored inside the Grid knowledge base. These services will be used both to build high-level knowledge discovery applications, as in the case of the Knowledge discovery, and to enhance existing basic Grid services. Two examples of high level services needing distributed data mining functionalities and accessing distributed partitions of a knowledge base are, for example, an enhanced version of the GridFTP protocol that classifies GridFTP usage data with data mining techniques to choose best connection parameters, and a Grid-based document management application.

Context-aware access to information and service adaptation. Semantic compression and synthesis of Grid information could be used to offer different views of the Grid knowledge base depending on many factors: user/service goals, scope of resource information. Other than usual compression techniques contents can be reorganized according to some aggregation functions, resulting in a synthetic yet meaningful version. Synthesis techniques, e.g. based on Data Mining metadata exploration, could enable the provision of different views of Grid resources exposing different levels of details, allowing adapting the access and use of such information to different users/services goals. Moreover, adaptation techniques coming from the Adaptive Hypermedia research community could be employed to adapt services to the user's computing environment on the basis of context.

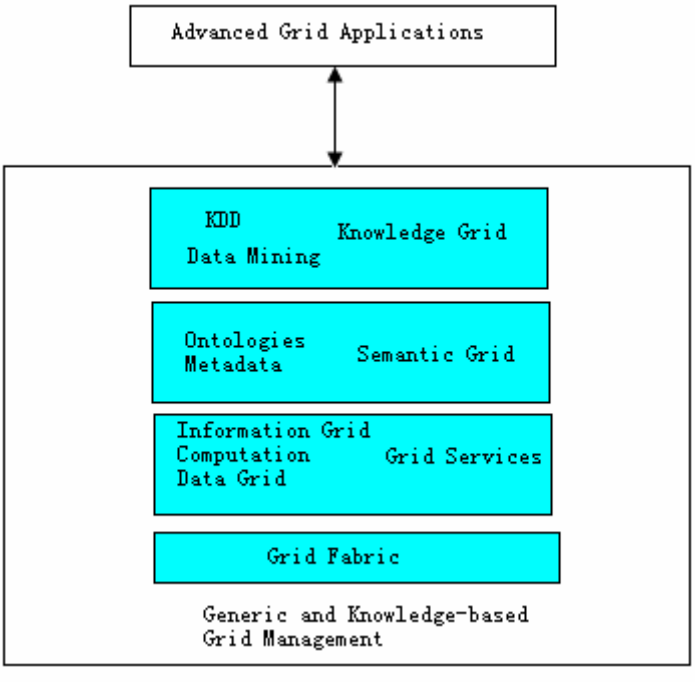


Fig. 1. Next-generation Grids Frame

Dynamic resource discovery. When the Grid goes beyond a static, well established configuration, becoming a Pervasive Grid, i.e. when new devices and resources are allowed to enter and exit the Grid in a very dynamic way, new services able to adapt themselves to the environment have to be developed. Peer-to-Peer technologies could be used to implement dynamic discovery algorithms.

Such services can be incrementally built leveraging current Grid efforts and projects. Figure 1 show how the recent research initiatives in the Grid community could be composed to provide a coherent architecture of services. Although these initiatives present some overlapping, they complement each others. Some enabling technologies, such as ontologies and reasoning, knowledge management and knowledge discovery, are currently offered by the depicted layers, but their main impact will be really evident when they will be used internally to enhance Grid management and operation. On the other hand, peer-to-peer and ubiquitous computing techniques start to be used very recently [4]. In our opinion peer-to-peer will be the orthogonal technology on which main tasks such as presence management, resource discovery and sharing, collaboration and self-configuration will be based.

3 Assessment Method

The fuzzy assessment method consists of the following steps: generate the objective and subjective assessment criteria, objectively assess the Next-generation Grids, subjectively assess the Next-generation Grids; integrate the objective and subjective assessment result according to the assessment value and the progressive weight distribution function, overall performance assessment.

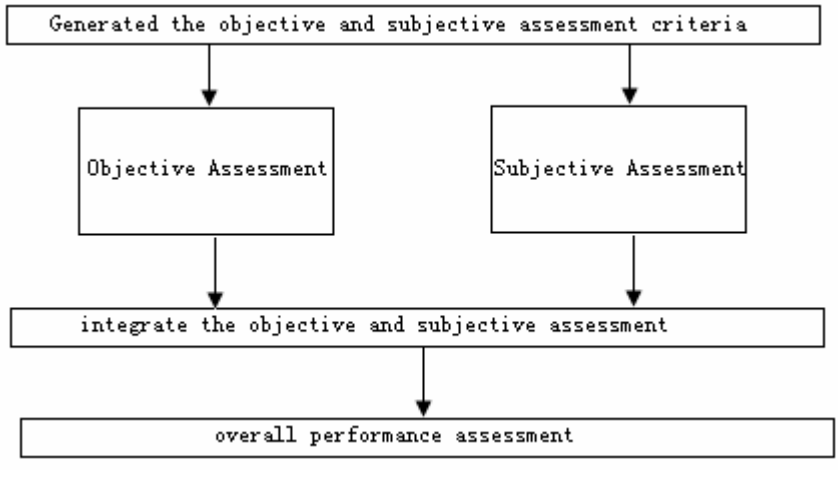


Fig. 2. Fuzzy assessment method for Next-generation Grids

3.1 Generate the Assessment Criteria

The Next-generation Grid developers propose the initial criterion set, and experts are invited to vote for the criteria online. The initial criteria will be filtered according to

the weighted average preferences and a predefined threshold. The new criterion set consists of the reserved criteria and those suggested by the experts. Advice and suggestions of the experts make the assessment criteria more complete and reasonable. After generating the new assessment criterion set, the experts are invited to revote and rank the criteria in a descending order. As a result, the criteria with high ranks construct the final assessment criterion set.

3.2 Objective Assessment of Next-Generation Grid

The objective Next-generation Grid Next-generation Grid assessment criteria are composed of knowledge organization, knowledge management, and technical aspects. The objective criteria related to knowledge organization include: the knowledge coverage degree, ratio of redundant semantic concern links, and ratio of inconsistent semantic links. The knowledge coverage degree of the Next-generation Grid reflects the connectivity of the stored knowledge. It can be computed as follows [5]:

$$K_C = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{m}\sum_{j=1}^{p}\frac{N_{Cj}}{N_{Fj}} \tag{1}$$

Where $\frac{1}{m}\sum_{j=1}^{p}\frac{N_{Cj}}{N_{Fj}}$ reflects the average knowledge coverage degree of network i and $\frac{N_{Cj}}{N_{Fj}} \geq \alpha$, $\alpha \in [0,1]$ the knowledge nodes in clique j of network i, N_{Fj} the number of knowledge nodes in fragment j of network i, $p \in Z^+$,and $p \leq m$ (the number of fragments in network i), and n the number of semantic link networks in the Next-generation Grid.

The criteria related to the knowledge management are assessed as follows: the consistency reflects the content correctness of the Next-generation Grid and can be computed by the ratio of the number of consistent knowledge nodes to the total number. The conciseness is assessed according to the ratio of the number of non-redundant knowledge nodes to the total number. The active nodes are the nodes that are activated in a certain frequency and used by the users or agents. The bigger the ratio of the number of active nodes to the total number, the better the performance of the Next-generation Grid is. The up-to-date degree is the ratio of the newly updated knowledge nodes to the total knowledge nodes. The knowledge category coverage degree reflects the ratio of categories in the Next-generation Grid to the available knowledge categories. The knowledge level coverage degree reflects the ratio of levels in the Next-generation Grid to the standard knowledge level classification. The satisfaction of users' interests reflects the average accordance of actual knowledge nodes distribution to the distribution of users' interests, which can be computed as follows [6]:

$$K_S = \frac{1}{n}\sum_{i=1}^{n}(1-|\alpha_{di}-\beta_{di}|) \tag{2}$$

Where $|\alpha_{di} - \beta_{di}|$ is the absolute value of the differences between α_{di} (the ratio of users who are interested in category d_i to the total users) and β_{di} (the ratio of the number of knowledge nodes about category d_i to the total number of knowledge nodes), and n the total number of the categories.

The technical assessment criteria include the response time and the across platform understandability.

3.3 Subjectively Assess the Next-Generation Grid

We use $V_{1\times n} = (p_{ij})^T$ $(j = 1,\cdots,n)$ to denote the preference vector of expert I for the n subjective criteria. The weighted average and variance of the preferences are considered when computing the subjective assessment value.

Let $P = (P_{1j},\cdots P_{mj})^T$ be the subjective assessment vector of m experts for criterion j, $W = (W_1,\cdots,W_m)^T$ be the weight vector. The overall subjective assessment for criterion j can be computed by

$$\begin{aligned} \mu_j &= \alpha \times \bar{X}_j - \beta \times S_n^2 \\ &= \alpha \times \sum_{i=1}^{m} P_{ij} \times W_i - \beta \times \frac{1}{m-1} \sum_{i=1}^{m} (P_{ij} - \bar{X}_j)^2 \end{aligned} \tag{3}$$

Where $\bar{X}_j = \sum_{i=1}^{m} P_{ij} \times W_i$ is the weighted average, $S_n^2 = \frac{1}{m-1} \sum_{i=1}^{m} (P_{ij} - \bar{X}_j)^2$ the variance, and $\alpha \in [0,1]$, $\beta \in [0,1]$ (the weight of the two) satisfy $\alpha + \beta = 1$.

3.4 Integrate the Objective and Subjective Assessment

Let $C = \{c_1, c_2, \cdots, c_n\}$ be the final assessment criterion set, the objective and subjective assessment value construct a K-measurable function $h(c_i)$, and the progressive weight distribution function $HW(c_i) = \sum_{k=1}^{i} w_k$ is 0-fuzzy metric. By using integral approach, the overall assessment of Next-generation Grid can be computed by

$$u = \max(\sum_{i=1}^{n} \min(h(c_i), Hw(c_i)) \tag{4}$$

3.5 Overall Performance Assessment for the Next-Generation Grid

According to the experts' suggestions, five fuzzy grades: Excellent, Good, Fair, P Poor, very Poor and the corresponding membership functions are established. The membership functions of the integrated assessment value indicate the overall performance of the Next-generation Grid.

4 Conclusions

The characteristics and structure of Next-generation Grids is proposed in the paper. The assessment of Next-generation Grids is composed of objective and subjective assessment. The method makes use of the fuzzy method to realize assessment and plan the improvement. The results manifest that the assessment method is effective through testing in some system. I wish that this article's work could give references to certain people.

References

1. Berman, F.: From TeraGrid to Knowledge Grid. CACM 44(11), 27–28 (2001)
2. Lyytinen, K., Yoo, Y.: Issues and Challenges in Ubiquitous Computing. CACM 45(12), 62–65 (2002)
3. Schoder, D., Fischbach, K.: Peer-to-Peer Prospects. CACM 46(2), 27–29 (2003)
4. Talia, D., Trunfio, P.: Toward a Synergy between P2P and Grids. IEEE Internet Computing 7(4), 94–96 (2003)
5. Zhuge, H.: A knowledge grid model and platform for global knowledge sharing, Expert System, pp. 313–320 (2002)
6. Zhuge, H.: Clustering soft-devices in semantic grid. IEEE Computing 4(6), 60–63 (2002)

Combining Prioritized Decisions in Classification

Yaxin Bi[1], Shengli Wu[1], and Gongde Guo[2]

[1] School of Computing and Mathematics
University of Ulster at Jordanstwon
Newtownabbey, Co. Antrim, BT37 0QB, UK
{y.bi, s.wu}@ulster.ac.uk
[2] Department of Computer Science
Fujian Normal University
Fuzhou, 350007, China
ggd@fjnu.edu.cn

Abstract. In this paper we present an alternative evidential method of combining prioritized decisions, in order to arrive at a "consensus", or aggregate, decision. Previous studies have suggested that, in some classification domains, the better performance can be achieved through combining the first and second decisions from each evidence source. However, it is easy to illustrate the fact that going further down a decision list, to give longer preferred decisions, can provide the alternative to the method of combining only the first one and second decisions. Our objective here is to examine the theoretical aspect of an alternative method in terms of *quartet* − how extending a decision list of any length by one extra preferred decision affects classification results. We also present the experimental results to demonstrate the effectiveness of our alternative method.

1 Introduction

There has been much interest from a wide range of areas in the representation of prioritized decisions/preferences and their subsequent manipulation. Applications have driven this to a great extent — for example, in areas of searching, optimization, planning, recommender systems, and efficient information access and management systems. Issues such as elicitation (or mining), representation and modelling, combination and merging of preferences, decisions and evidence have been extensively reviewed and addressed in [1].

The approach presented in this paper is based on the Dempster-Shafer (DS) theory of evidence, and it is applicable in many of the application domains listed above. We follow the Barnett's idea to use the DS theory of evidence to combine prioritized decisions which are naturally expressed as preferences for subsets of possible *decisions* [2]. We propose an approach, a model and theorems underpinning combination algorithms for this. Preference lists are combined to give aggregated decisions which can be used effectively in classification.

In this study our focus is upon *trimming* a list of prioritized decisions to restrict the length of the lists in each case − to 2 and 3 prioritized decisions − in order to transfer the DS theory to practical applications, and also to examine their effectiveness in classification. We describe novel structures, called *triplets* and *quartets*,

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 121–132, 2007.

for the representation of decisions in the truncated form, based on the stated or derived confidence levels associated with the decisions and organize the trimmed them into triplets, quartets, etc. depending on the number of list elements.

A previous study of applying the triplet-based approach to Text Categorization suggested that combining the best and the second best classification decisions from each classifier using the evidential operations can achieve good performance and efficiency [3] [4]. Clearly theory and methods are required for choosing the list length such as 3 decisions. Here we present detailed and general analysis of the quartet-based method, including the formal definition of *quartet*, the representation of prioritized decisions by quartets, the combination methods for multiple quartet mass function in any application areas, making available an alternative combining method in classification.

The rest of the paper is organized below. In section 2, we present an overview of the Dempster-Shafer (DS) theory of evidence. In Section 3, we introduce an application-specific mass function along with the triplet combination method. In section 4, we discuss the quartet definition and theoretical underpinning for establishing the algorithms of combining multiple quartet mass functions. In section 5, we present evaluation results based on 20-newsgroups − a benchmark data. Finally, we draw conclusions in Section 6.

2 An Overview of the Dempster-Shafer (DS) Theory of Evidence

In any exercise where decisions or preferences are to be combined, quantitative and qualitative pertinent information and knowledge often originate from different evidence sources and they are often pervaded with uncertainty. We seek a way to formalize the reasoning and decision processes − how evidence pertinent to a situation from multiple sources, is combined.

One method of combining evidence is to use the Dempster's rule or the *orthogonal sum*. It can make use of both judgemental and observational information available to find the 'best supported' decision [5]

Definition 1. Let Θ be a finite nonempty set, and call it the *frame of discernment*. Let $[0,1]$ denote the interval of real numbers from zero to one, inclusive: $[0,1] = \{x | 0 \leq x \leq 1\}$. A function $m : 2^{\Theta} \rightarrow [0,1]$ is called a *mass function* if it satisfies: $m(\emptyset) = 0$, $\sum_{X \subseteq \Theta} m(X) = 1$.

A mass function is a basic probability assignment *bpa* to all subsets X of Θ. A subset A of a frame Θ is called a *focal element* of a mass function m over Θ if $m(A) > 0$. These corresponding definitions apply to propositions as well as subsets.

The fundamental operation of evidential reasoning, namely, the orthogonal sum of evidential functions, is known as the Dempster's rule. Let m_1 and m_2 be mass functions on the frame Θ. Denote $N = \sum_{X \cap Y \neq \emptyset} m_1(X) m_2(Y)$. Suppose $N > 0$, i.e., $\sum_{X \cap Y = \emptyset} m_1(X) m_2(Y) < 1$. Then the following function $m : 2^{\Theta} \rightarrow [0,1]$ is a mass function: $m(\emptyset) = 0$, and $m(A) = \sum_{X \cap Y = A} m_1(X) m_2(Y)/N$ for all subsets $A \neq \emptyset$ of Θ. The mass function m is called the orthogonal sum of m_1

and m_2, and is denoted by $m_1 \oplus m_2$, and $K = 1/N$ is called the *normalization constant* of the orthogonal sum of m_1 and m_2. If $N = 0$, then we say that the orthogonal sum $m_1 \oplus m_2$ *does not exist*, and that m_1 and m_2 are totally contradictory. We often allocate some mass to undeterministic status by means of *ignorance*.

3 The Triplet Combination Method in Classification

We consider a generic approach to classification. Suppose we are given a classification model, for new instances, the process of classifying any instance is to calculate similarity scores between a instance and the model. It is intuitively sensible that the higher this score, the more likely the instance belongs to the class to which the model dedicates. Formally let $D = \{d_1, d_2, ..., d_{|D|}\}$ be a training set of instances, where d is represented by a $|V|$-dimensional weighted vector and V is a set of attribute / feature values. Let $C = \{c_1, \cdots, c_{|C|}\}$ be a set of categories, then the task of assigning predefined categories to instances can be regarded as a mapping of each pair $< d, c > \in D \times C$ to a boolean value true (T) or false (F). If value T is assigned to $< d, c >$, it means that a decision is made to include instance d under category c, whereas value F indicates that instance d is not under category c. A learning for classification task is to construct an approximation to an unknown function $\varphi : D \times C \rightarrow \{T, F\}$, where φ is called a *classifier*.

However, given a test instance d, such a mapping cannot guarantee that an assignment of a category to d is either true or false; instead we use $|C|$-dimensional vector of numeric values, denoted by $\varphi(d) = \{s_1, \cdots, s_{|C|}\}$, where these s_i represent the relevance of document d to the respective categories in the form of similarity scores or probabilities, i.e. the greater the score for category c_i, the greater the correspondence of the document to that category. Now we model a $|C|$-dimensional vector of numeric values as a preference by prioritizing these values.

Definition 2. Let C be a frame of discernment and D be a training set of instances, where each choice $c_i \in C$ is a proposition of the form *source* $d \in D$ *chooses, or supports* c_i, and let $\varphi(d)$ be a piece of evidence to allocate a strength s_i to each choice $c_1, \cdots, c_{|C|}$.

Then $m(\{c_i\}) = \frac{s_i}{\sum_{k=1}^{|C|} s_k}$ is a *basic probability assignment* (*bpa*) to c_i for $1 \leq i \leq |C|$. It is also a mass function *bpa* that expresses the degrees of beliefs in respective propositions corresponding to each choice to which a given source could belong.

Definition 3. Let C be a frame of discernment and $\varphi(d) = (m(\{c_1\}), \cdots, m(\{c_{|C|}\}))$, where $|C| \geq 2$, a *focal element triplet* is defined as an expression of the form $Y = \langle A_1, A_2, A_3 \rangle$, where $A_1, A_2 \subseteq C$ are singleton, and A_3 is the whole set C.

To allocate mass values to A_1, A_2 and A_3, we rank $\varphi(d)$ to be $m(c_{i_1}) \geq m(c_{i_2}) \geq ... \geq m(c_{i_{|C|}})$ in decreasing order. Then we have prioritized decisions $A_1 = \{c_{i_1}\}, A_2 = \{c_{i_2}\}$ alone with $A_3 = C$. The associated mass function is:

$m(A_1) = m(\{c_{i_1}\})$, $m(A_2) = m(\{c_{i_2}\})$ and $m(A_3) = 1 - m(A_1) - m(A_2)$ that represents *ignorance*. The mass function in such a form is called *triplet mass function, 2-points mass function, or simply 2-points focussed*, an important property associated with the triplet mass function is that it has no focuses other than two singletons and Θ. That is, there exist two and only two elements $x, y \in \Theta$ such that

$$m(\{x\}) + m(\{y\}) + m(\Theta) = 1;\ 0 \leq m(\{x\}), m(\{y\}), m(\Theta) \leq 1.$$

4 Focusing on 3 Focal Elements — Quartet

Generally, a mass function may have more than three focal singletons. We have briefly presented 2 points focussed in the previous section, the detailed discussions can be found in [4], [6]. Similarly, we can consider 3-points focussed, 4-points focussed, ..., n-points focussed mass functions. We here confine our interest in 3-points focussed − quartet. Now we give a formal definition of three-points focused mass function

Definition 4. Let C be a frame of discernment and $\varphi(d) = (m(\{c_1\}), \cdots, m(\{c_{|C|}\}))$, where $|C| \geq 2$, a *quartet* is defined as an expression of the form $Y = \langle A_1, A_2, A_3, A_4 \rangle$, where $A_1, A_2, A_3 \subseteq C$ are singleton, and A_4 is the whole set C.

In order to organize a list of decisions into a quartet, we again rank $\varphi(d)$ to be $m(c_{i_1}) \geq m(c_{i_2}) \geq ... \geq m(c_{i_{|C|}})$ in decreasing order and then trims down the prioritized decisions to three points focused, as a result we have $A_1 = \{c_{i_1}\}, A_2 = \{c_{i_2}\}, A_3 = \{c_{i_3}\}$ and $A_4 = C$. The associated mass function is: $m(A_1) = m(\{c_{i_1}\})$, $m(A_2) = m(\{c_{i_2}\})$, $m(A_3) = m(\{c_{i_3}\})$and $m(A_4) = 1 - m(A_1) - m(A_2) - m(A_3)$

We have looked at the computational process for combining triplet mass functions [6]. Given two quartets $A = \langle A_1, A_2, A_3, A_4 \rangle$ and $B = \langle B_1, B_2, B_3, B_4 \rangle$ and associated mass functions m_1 and m_2, we need to consider enumerative relations between any pairs of singletons from A and B to compute combinations of two quartet mass functions. Here we identify situations which two mass functions in quartet form can be combined together for the cases where all 3 focuses are equal, 2 focuses are equal, when just one focus is equal, and when there are no focus are in common, respectively. The corresponding analysis addresses, in each case, the situation where the focuses are ordered by their masses for each of the two pieces of evidence being combined, which have been implemented in a set of algorithms.

Now we develop four theorems ensuring that the combinations of quartet mass functions exist for the different cases stated above. These theorems form the theoretical basis for establishing a set of formulae for computing various quartet mass functions.

First we consider the case where the 3 focuses are equal (eg x, y, z in the following context).

Theorem 1. Let m_1, m_2 be two 3-points focussed mass functions having 3 points equal,

$$m_1(\{x\}) + m_1(\{y\}) + m_1(\{z\}) + m_1(\Theta) = 1;\ 0 \leq$$
$$m_1(\{x\}), m_1(\{y\}), m_1(\{z\}), m_1(\Theta) \leq 1;$$
$$m_2(\{x\}) + m_2(\{y\}) + m_2(\{z\}) + m_2(\Theta) = 1;\ 0 \leq$$
$$m_2(\{x\}), m_2(\{y\}), m_2(\{z\}), m_2(\Theta) \leq 1.$$

Then $1/K = N = 1 - m_1(\{x\})m_2(\{y\}) - m_1(\{y\})m_2(\{x\})$

$$-m_1(\{x\})m_2(\{z\}) - m_1(\{z\})m_2(\{x\}) - m_1(\{y\})m_2(\{z\}) - m_1(\{z\})m_2(\{y\}),$$

and m_1, m_2 are combinable if and only if

$$m_1(\{x\})m_2(\{y\}) + m_1(\{y\})m_2(\{x\}) + m_1(\{x\})m_2(\{z\}) + m_1(\{z\})m_2(\{x\})$$
$$+m_1(\{y\})m_2(\{z\}) + m_1(\{z\})m_2(\{y\}) < 1.$$

When m_1, m_2 are combinable we have

$$(m_1 \oplus m_2)(\{x\}) = K(m_1(\{x\})m_2(\{x\}) + m_1(\{x\})m_2(\Theta) + m_1(\Theta)m_2(\{x\})), \quad (1)$$
$$(m_1 \oplus m_2)(\{y\}) = K(m_1(\{y\})m_2(\{y\}) + m_1(\{y\})m_2(\Theta) + m_1(\Theta)m_2(\{y\})), \quad (2)$$
$$(m_1 \oplus m_2)(\{z\}) = K(m_1(\{z\})m_2(\{z\}) + m_1(\{z\})m_2(\Theta) + m_1(\Theta)m_2(\{z\})), \quad (3)$$
$$(m_1 \oplus m_2)(\Theta) = Km_1(\Theta)m_2(\Theta). \quad (4)$$

Proof. By using the orthogonal sum operation to combine m_1, m_2, we can obtain $(m_1 \oplus m_2)(A) = K \sum_{X \cap Y = A} m_1(X)m_2(Y)$, where we have that

$$1/K = N = 1 - \sum_{X \cap Y = \emptyset} m_1(X)m_2(Y) = 1 - m_1(\{x\})m_2(\{y\}) - m_1(\{y\})m_2(\{x\})$$

$$-m_1(\{x\})m_2(\{z\}) - m_1(\{z\})m_2(\{x\}) - m_1(\{y\})m_2(\{z\}) - m_1(\{z\})m_2(\{y\}),$$

and it is easy to see that m_1, m_2 are combinable if and only if $N > 0$, i.e.,

$$1 - m_1(\{x\})m_2(\{y\}) - m_1(\{y\})m_2(\{x\}) - m_1(\{x\})m_2(\{z\}) - m_1(\{z\})m_2(\{x\})$$
$$-m_1(\{y\})m_2(\{z\}) - m_1(\{z\})m_2(\{y\}) > 0$$

i.e., $m_1(\{x\})m_2(\{y\}) + m_1(\{y\})m_2(\{x\}) + m_1(\{x\})m_2(\{z\}) + m_1(\{z\})m_2(\{x\})$
$$+m_1(\{y\})m_2(\{z\}) + m_1(\{z\})m_2(\{y\}) < 1.$$

By the orthonogonal sum operation above for $m_1 \oplus m_2$ we find that

$$(m_1 \oplus m_2)(\{x\}) = K(m_1(\{x\})m_2(\{x\}) + m_1(\{x\})m_2(\Theta) + m_1(\Theta)m_2(\{x\})),$$
$$(m_1 \oplus m_2)(\{y\}) = K(m_1(\{y\})m_2(\{y\}) + m_1(\{y\})m_2(\Theta) + m_1(\Theta)m_2(\{y\})),$$

$$(m_1 \oplus m_2)(\{z\}) = K(m_1(\{z\})m_2(\{z\}) + m_1(\{z\})m_2(\Theta) + m_1(\Theta)m_2(\{z\})),$$

$$(m_1 \oplus m_2)(\Theta) = Km_1(\Theta)m_2(\Theta). \quad \square$$

Now the 3 points equal case is probably the simplest. We need to consider other possibilities.

Theorem 2. Let m_1, m_2 be two 3-points focussed mass functions having two equal points,

$$m_1(\{x\}) + m_1(\{y\}) + m_1(\{z\}) + m_1(\Theta) = 1;\ 0 \leq$$

$$m_1(\{x\}), m_1(\{y\}), m_1(\{z\}), m_1(\Theta) \leq 1;$$

$$m_2(\{x\}) + m_2(\{y\}) + m_2(\{u\}) + m_2(\Theta) = 1;\ 0 \leq$$

$$m_2(\{x\}), m_2(\{y\}), m_2(\{u\}), m_2(\Theta) \leq 1.$$

Then $1/K = N = 1 - m_1(\{x\})m_2(\{y\}) - m_1(\{y\})m_2(\{x\}) - m_1(\{z\})m_2(\{u\})$

$$-m_1(\{x\})m_2(\{u\}) - m_1(\{z\})m_2(\{x\}) - m_1(\{y\})m_2(\{u\}) - m_1(\{z\})m_2(\{y\}),$$

and m_1, m_2 are combinable if and only if

$$m_1(\{x\})m_2(\{y\}) + m_1(\{y\})m_2(\{x\}) + m_1(\{x\})m_2(\{u\}) + m_1(\{z\})m_2(\{x\})$$

$$+m_1(\{y\})m_2(\{u\}) + m_1(\{z\})m_2(\{y\}) + m_1(\{z\})m_2(\{u\}) < 1.$$

When m_1, m_2 are combinable we have

$$(m_1 \oplus m_2)(\{x\}) = K(m_1(\{x\})m_2(\{x\}) + m_1(\{x\})m_2(\Theta) + m_1(\Theta)m_2(\{x\})), \quad (5)$$

$$(m_1 \oplus m_2)(\{y\}) = K(m_1(\{y\})m_2(\{y\}) + m_1(\{y\})m_2(\Theta) + m_1(\Theta)m_2(\{y\})), \quad (6)$$

$$(m_1 \oplus m_2)(\{z\}) = Km_1(\{z\})m_2(\Theta), \quad (7)$$

$$(m_1 \oplus m_2)(\{u\}) = Km_1(\Theta)m_2(\{u\}), \quad (8)$$

$$(m_1 \oplus m_2)(\Theta) = Km_1(\Theta)m_2(\Theta). \quad (9)$$

Proof

To combine m_1, m_2 by using the orthogonal sum operation, we have $(m_1 \oplus m_2)(A) = K \sum_{X \cap Y = A} m_1(X)m_2(Y)$, where $1/K = N$

$$= 1 - \sum_{X \cap Y = \emptyset} m_1(X)m_2(Y) = 1 - m_1(\{x\})m_2(\{y\}) - m_1(\{y\})m_2(\{x\})$$

$$-m_1(\{z\})m_2(\{u\})$$

$$-m_1(\{x\})m_2(\{u\}) - m_1(\{z\})m_2(\{x\}) - m_1(\{y\})m_2(\{u\}) - m_1(\{z\})m_2(\{y\}),$$

and it is straightforward to obtain that m_1, m_2 are combinable if and only if $N > 0$, i.e.,

$$1 - m_1(\{x\})m_2(\{y\}) - m_1(\{y\})m_2(\{x\}) - m_1(\{z\})m_2(\{u\})$$

$$-m_1(\{x\})m_2(\{u\})-m_1(\{z\})m_2(\{x\})-m_1(\{y\})m_2(\{u\})-m_1(\{z\})m_2(\{y\})>0$$

i.e., $m_1(\{x\})m_2(\{y\})+m_1(\{y\})m_2(\{x\})+m_1(\{x\})m_2(\{u\})+m_1(\{z\})m_2(\{x\})$

$$+m_1(\{y\})m_2(\{u\})+m_1(\{z\})m_2(\{y\})+m_1(\{z\})m_2(\{u\})<1.$$

Similarly, by the orthogonal sum formula for $m_1 \oplus m_2$ we find that

$$(m_1 \oplus m_2)(\{x\}) = K(m_1(\{x\})m_2(\{x\}) + m_1(\{x\})m_2(\Theta) + m_1(\Theta)m_2(\{x\})),$$

$$(m_1 \oplus m_2)(\{y\}) = K(m_1(\{y\})m_2(\{y\}) + m_1(\{y\})m_2(\Theta) + m_1(\Theta)m_2(\{y\})),$$

$$(m_1 \oplus m_2)(\{z\}) = Km_1(\{z\})m_2(\Theta), (m_1 \oplus m_2)(\{u\}) = Km_1(\Theta)m_2(\{u\}),$$

$$(m_1 \oplus m_2)(\Theta) = Km_1(\Theta)m_2(\Theta). \qquad \square$$

The following theorem guarantees two quartet mass functions are combinable when only one focus might be shared by the two sets of results.

Theorem 3. Let m_1, m_2 be two 3-points focussed mass functions having one equal point,

$$m_1(\{x\}) + m_1(\{y\}) + m_1(\{z\}) + m_1(\Theta) = 1;$$

$$0 \leq m_1(\{x\}), m_1(\{y\}), m_1(\{z\}), m_1(\Theta) \leq 1;$$

$$m_2(\{x\}) + m_2(\{u\}) + m_2(\{v\}) + m_2(\Theta) = 1;$$

$$0 \leq m_2(\{x\}), m_2(\{u\}), m_2(\{v\}), m_2(\Theta) \leq 1.$$

Then $1/K = N = 1 - m_1(\{x\})m_2(\{u\}) - m_1(\{y\})m_2(\{x\}) - m_1(\{z\})m_2(\{v\}$

$$-m_1(\{y\})m_2(\{u\}) - m_1(\{x\})m_2(\{v\}) - m_1(\{z\})m_2(\{x\}) - m_1(\{y\})m_2(\{v\})$$

$$-m_1(\{z\})m_2(\{u\}),$$

and m_1, m_2 are combinable if and only if

$$m_1(\{x\})m_2(\{u\}) + m_1(\{y\})m_2(\{x\}) + m_1(\{x\})m_2(\{v\}) + m_1(\{z\})m_2(\{x\})$$

$$+m_1(\{y\})m_2(\{v\})+m_1(\{z\})m_2(\{u\})+m_1(\{z\})m_2(\{v\}+m_1(\{y\})m_2(\{u\})<1.$$

When m_1, m_2 are combinable we have

$$(m_1 \oplus m_2)(\{x\}) = K(m_1(\{x\})m_2(\{x\}) + m_1(\{x\})m_2(\Theta) + m_1(\Theta)m_2(\{x\})), \tag{10}$$

$$(m_1 \oplus m_2)(\{y\}) = Km_1(\{y\})m_2(\Theta), \tag{11}$$

$$(m_1 \oplus m_2)(\{z\}) = Km_1(\{z\})m_2(\Theta), \tag{12}$$

$$(m_1 \oplus m_2)(\{u\}) = Km_1(\Theta)m_2(\{u\}), \tag{13}$$

$$(m_1 \oplus m_2)(\{v\}) = Km_1(\Theta)m_2(\{v\}), \tag{14}$$

$$(m_1 \oplus m_2)(\Theta) = Km_1(\Theta)m_2(\Theta). \tag{15}$$

Proof
To combine m_1, m_2 by using the orthogonal sum operation, we know that $(m_1 \oplus m_2)(A) = K \sum_{X \cap Y = A} m_1(X) m_2(Y)$,
where $1/K = N = 1 - \sum_{X \cap Y = \emptyset} m_1(X) m_2(Y)$

$$= 1 - m_1(\{x\})m_2(\{u\}) - m_1(\{y\})m_2(\{x\}) - m_1(\{z\})m_2(\{v\} - m_1(\{y\})m_2(\{u\})$$

$$-m_1(\{x\})m_2(\{v\}) - m_1(\{z\})m_2(\{x\}) - m_1(\{y\})m_2(\{v\}) - m_1(\{z\})m_2(\{u\}),$$

from this expression, we can see the statement of that m_1, m_2 are combinable is true if and only if with the fact of $N > 0$, i.e.,

$$1 - m_1(\{x\})m_2(\{u\}) - m_1(\{y\})m_2(\{x\}) - m_1(\{z\})m_2(\{v\} - m_1(\{y\})m_2(\{u\})$$

$$-m_1(\{x\})m_2(\{v\}) - m_1(\{z\})m_2(\{x\}) - m_1(\{y\})m_2(\{v\}) - m_1(\{z\})m_2(\{u\}) > 0$$

i.e.,

$$m_1(\{x\})m_2(\{u\}) + m_1(\{y\})m_2(\{x\}) + m_1(\{x\})m_2(\{v\}) + m_1(\{z\})m_2(\{x\})$$

$$+m_1(\{y\})m_2(\{v\}) + m_1(\{z\})m_2(\{u\}) + m_1(\{z\})m_2(\{v\} + m_1(\{y\})m_2(\{u\}) < 1.$$

By the orthogonal sum for $m_1 \oplus m_2$ we find that

$$(m_1 \oplus m_2)(\{x\}) = K(m_1(\{x\})m_2(\{x\}) + m_1(\{x\})m_2(\Theta) + m_1(\Theta)m_2(\{x\})),$$

$$(m_1 \oplus m_2)(\{y\}) = K m_1(\{y\})m_2(\Theta), (m_1 \oplus m_2)(\{z\}) = K m_1(\{z\})m_2(\Theta),$$

$$(m_1 \oplus m_2)(\{u\}) = K m_1(\Theta)m_2(\{u\}), (m_1 \oplus m_2)(\{v\}) = K m_1(\Theta)m_2(\{v\}),$$

$$(m_1 \oplus m_2)(\Theta) = K m_1(\Theta)m_2(\Theta). \qquad \square$$

In the following we consider the case where no any focus is shared between the two sets of results.

Theorem 4. Let m_1, m_2 be two 3-points focussed mass functions having no equal point,

$$m_1(\{x\}) + m_1(\{y\}) + m_1(\{z\}) + m_1(\Theta) = 1$$

$$0 \leq m_1(\{x\}), m_1(\{y\}), m_1(\{z\}), m_1(\Theta) \leq 1;$$

$$m_2(\{u\}) + m_2(\{v\}) + m_2(\{w\}) + m_2(\Theta) = 1$$

$$0 \leq m_2(\{u\}), m_2(\{v\}), m_2(\{w\}), m_2(\Theta) \leq 1.$$

Then

$$1/K = N = 1 - m_1(\{x\})m_2(\{u\}) - m_1(\{y\})m_2(\{u\}) - m_1(\{z\})m_2(\{u\}$$

$$-m_1(\{x\})m_2(\{v\})$$

$$-m_1(\{y\})m_2(\{v\}) - m_1(\{z\})m_2(\{v\} - m_1(\{x\})m_2(\{w\}) - m_1(\{y\})m_2(\{w\})$$

$$-m_1(\{z\})m_2(\{w\},$$

and m_1, m_2 are combinable if and only if

$$m_1(\{x\})m_2(\{u\}) + m_1(\{y\})m_2(\{u\}) + m_1(\{z\})m_2(\{u\}) +$$

$$m_1(\{x\})m_2(\{v\}) + m_1(\{y\})m_2(\{v\}) + m_1(\{z\})m_2(\{v\}) +$$

$$m_1(\{x\})m_2(\{w\}) + m_1(\{y\})m_2(\{w\}) + m_1(\{z\})m_2(\{w\}) < 1.$$

When m_1, m_2 are combinable we have

$$(m_1 \oplus m_2)(\{x\}) = Km_1(\{x\})m_2(\Theta), \tag{16}$$

$$(m_1 \oplus m_2)(\{y\}) = Km_1(\{y\})m_2(\Theta), \tag{17}$$

$$(m_1 \oplus m_2)(\{z\}) = Km_1(\{z\})m_2(\Theta), \tag{18}$$

$$(m_1 \oplus m_2)(\{u\}) = Km_1(\Theta)m_2(\{u\}), \tag{19}$$

$$(m_1 \oplus m_2)(\{v\}) = Km_1(\Theta)m_2(\{v\}), \tag{20}$$

$$(m_1 \oplus m_2)(\{w\}) = Km_1(\Theta)m_2(\{w\}), \tag{21}$$

$$(m_1 \oplus m_2)(\Theta) = Km_1(\Theta)m_2(\Theta). \tag{22}$$

Proof
To combine m_1, m_2, we need to consider the orthogonal sum operation for $m_1 \oplus m_2$, and we know that $(m_1 \oplus m_2)(A) = K \sum_{X \cap Y = A} m_1(X)m_2(Y)$, where we have that $1/K = N = 1 - \sum_{X \cap Y = \emptyset} m_1(X)m_2(Y)$

$$= 1 - m_1(\{x\})m_2(\{u\}) - m_1(\{y\})m_2(\{u\}) - m_1(\{z\})m_2(\{u\} - m_1(\{x\})m_2(\{v\})$$

$$-m_1(\{y\})m_2(\{v\}) - m_1(\{z\})m_2(\{v\} - m_1(\{x\})m_2(\{w\}) - m_1(\{y\})m_2(\{w\})$$

$$-m_1(\{z\})m_2(\{w\},$$

and clearly to guarantee that m_1, m_2 are combinable, it must be ensured that $N > 0$, i.e.,

$$1 - m_1(\{x\})m_2(\{u\}) - m_1(\{y\})m_2(\{u\}) - m_1(\{z\})m_2(\{u\}$$

$$-m_1(\{x\})m_2(\{v\}) - m_1(\{y\})m_2(\{v\}) - m_1(\{z\})m_2(\{v\}$$

$$-m_1(\{x\})m_2(\{w\}) - m_1(\{y\})m_2(\{w\}) - m_1(\{z\})m_2(\{w\} > 0$$

i.e.,

$$m_1(\{x\})m_2(\{u\}) + m_1(\{y\})m_2(\{u\}) + m_1(\{z\})m_2(\{u\}) +$$

$$m_1(\{x\})m_2(\{v\}) + m_1(\{y\})m_2(\{v\}) + m_1(\{z\})m_2(\{v\}) +$$

$$m_1(\{x\})m_2(\{w\}) + m_1(\{y\})m_2(\{w\}) + m_1(\{z\})m_2(\{w\}) < 1.$$

By the orthogonal sum operation above, we find that

$$(m_1 \oplus m_2)(\{x\}) = Km_1(\{x\})m_2(\Theta), (m_1 \oplus m_2)(\{y\}) = Km_1(\{y\})m_2(\Theta),$$

$$(m_1 \oplus m_2)(\{z\}) = Km_1(\{z\})m_2(\Theta), (m_1 \oplus m_2)(\{u\}) = Km_1(\Theta)m_2(\{u\}),$$
$$(m_1 \oplus m_2)(\{v\}) = Km_1(\Theta)m_2(\{v\}), (m_1 \oplus m_2)(\{w\}) = Km_1(\Theta)m_2(\{w\}),$$
$$(m_1 \oplus m_2)(\Theta) = Km_1(\Theta)m_2(\Theta). \quad \square$$

With M triplet mass functions existing, we can use the formula of pairwise orthogonal sums below (Equation 23) to combine them. However for each computational result, the focusing operator may be applied to the combination process proceed iteratively. Finally it produces a new mass distribution over some classes, which represents the consensus of the decisions of multiple classifiers to a new instance.

$$m_1 \oplus m_2 \oplus [m_3 \oplus [m_4 \oplus ...[m_{M-1} \oplus m_M]]...] \tag{23}$$

We have extended the triplet structure to greater numbers of focuses — quartet. With these theorems, we obtain the conditions where using an additional focus in decision list is desirable. This theoretical results underpins a set of algorithms by allowing the alternative structure length to be chosen for decision aggregations.

5 Evaluation

In this section we describe the experiment which has been performed to evaluate our combination method given in the previous sections. For our experiments, we have chosen a public benchmark dataset, often referred to as 20-newsgroup. It consists of 20 categories, and each category has 1,000 documents (Usenet articles), so the dataset contains 20,000 documents in total. Except for a small fraction of the articles (4%), each article belongs to exactly one category [7].

We use *information gain* as a measure for feature selection at the preprocessing stage for each classification method, and weight features by using *tfidf* (term frequency within the document and inverse document frequency) after removing function words and applying stemming [8]. 5300 features have been selected in total. The experiments have been conducted using a ten-fold cross validation. For each classification method, ten classifiers are generated and the performance of the method is the mean value of the ten classifiers. The performance of learning algorithms have been measured using a measure which is widely used in information retrieval and text categorization: the macro-average F_1 defined on a pair of measures, called Precision and Recall [9].

Fig. 1 demonstrates the performance comparison among the best combined classifier (SVM and kNNM - called SM) and 4 individual classifiers (SVM, kNNM, kNN and Rocchio) on 20 document categories. The best combined classifier outperforms any individual classifiers on the average. The estimated performance of the best combination is 90.15%, which is 2.69% better than the best individual classifier (SVM). Fig. 2 illustrates the performance comparison among the best combinations of two classifiers SM (SVM + kNNM), three classifiers SMR (SVM + kNNM + Rocchio), and the four classifiers SMNR (SVM + kNNM + kNN + Rocchio). As we see, the best combination of two classifiers

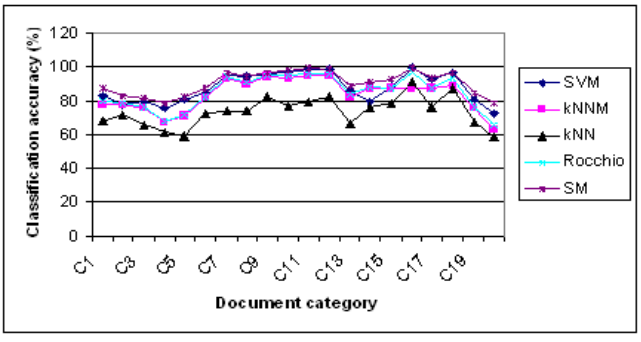


Fig. 1. The performance of the best combined classifier SM (SVM + kNNM) against the individual classifier SVM, kNNM, kNN, and Rocchio

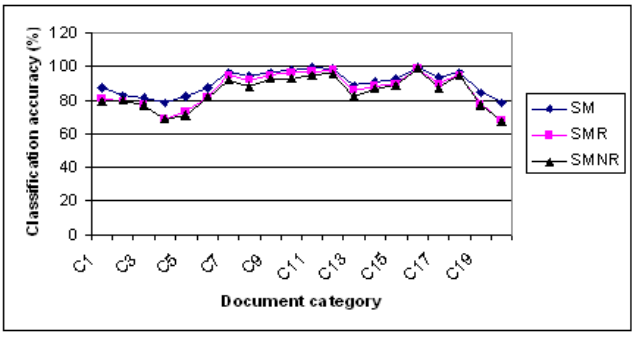


Fig. 2. The performance comparison of the combined classifiers SM (SVM + kNNM), SMR (SVM + kNNM + Rocchio) and SMNR (SVM + kNNM + kNN + Rocchio)

SM outperforms SMR and SMNR, and the performance of the best combination of SMR is almost the same as that of SMNR with the exception of document categories of 8 - 11 and 13 - 16. The estimated classification accuracies of SMR and SMNR are 86.12% and 84.58% respectively, which are 1.35% and 2.88% worse than the best individual classifier SVM. So our experimental results show that the combination of the best and the second best classifiers is the best combination that outperforms the individual classifiers and the combined classifiers. These experimental results are consistent with those presented in [3,6]

6 Conclusion

We suggest the extension of a novel technique triplet to quartet to greater number of focuses for representing prioritized decisions, and present an evidential method for combining multiple classification decisions based on this new evidence structure. In particular our experimental results show the performance of

the quartet method is comparable with that of the triplet method in combining classification decisions over the same benchmark data of 20-newsgroup. Like the triplet structure, this structure, and the associated methods and techniques developed in this research are particularly useful for preference analysis and decision making when knowledge and information are insufficient and incomplete.

References

1. Delgrande, J., Schaub, T., Tompits, H., Wang, K.: A classification and survey of preference handling approaches in nonmonotonic reasoning. Computational Intelligence 20(2), 308–334 (2004)
2. Barnett, J.: Combining opinions about the order of rule execution. In: AAAI, pp. 477–481(1991)
3. Bi, Y., Bell, D.A., Wang, H., Guo, G., Greer, K.: Combining multiple classifiers using dempster's rule of combination for text categorization. In: Torra, V., Narukawa, Y. (eds.) MDAI 2004. LNCS (LNAI), vol. 3131, pp. 127–138. Springer, Heidelberg (2004)
4. Bi, Y., Bell, D.A., Guan, J.: Combining evidence from classifiers in text categorization. In: Negoita, M.G., Howlett, R.J., Jain, L.C. (eds.) KES 2004. LNCS (LNAI), vol. 3215, pp. 521–528. Springer, Heidelberg (2004)
5. Shafer, G.: A Mathematical Theory of Evidence. Princeton University Press, Princeton, NJ (1976)
6. Bi, Y.: Combining Multiple Classifiers for Text Categorization using Dempster's rule of combination. PhD thesis, University of Ulster (2004)
7. Joachims, T.: A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In: ICML '97. Proceedings of the Fourteenth International Conference on Machine Learning, San Francisco, CA, USA, pp. 143–151. Morgan Kaufmann Publishers Inc. San Francisco (1997)
8. Salton, G., Allan, J., Buckley, C., Singhal, A.: Automatic analysis, theme generation, and summarization of machine-readable texts. Science 264, 1421–1426 (1994)
9. Rijsbergen, C.J.V.: Information Retrieval, 2nd edn. Dept. of Computer Science, University of Glasgow (1979)

An Aggregation of Agents, Roles and Coalition Formation to Support Collaborative and Dynamic Organizations

Nora Houari and Behrouz H. Far

Department of Electrical and Computer Engineering,
University of Calgary, 2500 University Drive N.W.
T2N 1N4 Alberta, Canada
{nhouari,far}@ucalgary.ca

Abstract. Capturing human resources, their capabilities and dynamics can greatly contribute to an efficient and effective organization mission goals accomplishment. In practice however, this is often done in an ad-hoc manner leading to a big gap between the organizational specification model (at the design time) and its instance model (during the life cycle). In this paper we present a generic organizational model that captures the human resources, their operational and strategic capabilities, and adaptive coalition formation. The key elements of the model are capabilities, roles, agents and coalitions. We show how the resulting model can be used for role functionalities and performance control, as well as for capturing the knowledge in context-aware applications which is often present in an implicit manner.

Keywords: adaptive organizational modeling, human in the loop, support system, operational activities, strategic goals, coalition, collaboration.

1 Introduction

It is now well-acknowledged that the field of multiagent systems is well suited for real world social structure settings, such as individuals, collaborative teams, and organization support. The design and use of any of these agent-based systems necessitates careful consideration of human resources, their roles, capabilities and environment adaptability. The importance of capturing human resources in the modeling process is obvious: first, human are characterized by their non-deterministic and subjective behavior which has a paramount impact on the organizational process evolution; second, the effect of the execution risks of activities which are not adequate to the capacity of human resources; third the importance of capturing their operational and strategic capabilities, and adaptability to changes in the environment to achieve the required goals' efficiency and effectiveness. Therefore, these reasons entail for an explicit examination of knowledge which is often present in an implicit manner. Examples of such knowledge are who is involved, who is doing what with whom, and how interactions are held.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 133–144, 2007.

In this paper we provide a formal approach for human centered adaptable organizational modeling which includes integrating physical human agents in the modeling loop, incorporation of both role's capabilities (operational and strategic) and coalitions formation. We identify four main pillars that have relevant impact on dynamic organizational modeling: (i) capabilities, (ii) roles, (iii) agents, and (iv) coalitions. The main idea of our approach is to integrate both of the identified capabilities (operational and strategic) in a coherent and unified form; our technical approach to this integration is based on building the coalition of agents for an organizational role and their adaptation for a conceptual role. The objective here is that human will be able to assign the performance management of their activities to software agents so that they can fully focus their cognitive capabilities on processing, decision making, etc.

The remainder of the paper is structured as follows: Section 2 describes the fundamental considerations for building our adaptive organizations approach. Section 3 presents in details our formally founded organization model. In Section 4 we delineate the system architecture, and in Section 5 we summarize the related work. Finally, Section 6 concludes and outlines the ongoing and future works.

2 Fundamental Considerations

Although there is a wide variability of what an organization is, in organizational research areas an organization refers to humans and operational functions that are distributed among team members who coordinate their actions in order to achieve a common goal [8]. The modeling of today's competitive organizations calls for integration of multiple work systems in a single information system that coordinates multiple tasks performed by multiple actors, who are individually neither capable nor knowledgeable about all the different tasks in the entire organizational environment. The underlying implementation of this integration is based on the organization's specifications, its instances, and dynamics. Thus, organizational modeling cannot be fixed in every detail at the specification level (building time); instead it should be possible to model the organization only at the coarse-grained level, and to allow an optimized evolution at organizational instance level according to its norms and specific goals.

In this research we consider an integrated organizational model that essentially incorporates capabilities structured into roles that are played by agents who participate in coalitions to achieve a common goal; that is an organization is a tuple of <capabilities, roles, agents, coalitions>. Figure 1 depicts key elements of our collaborative and dynamic organization model which allows integrating human in the loop, captures operational and strategic capabilities on one hand and their changes on the other. In detail, the model must:

- Provide human agents and teams with awareness of their work evolution with precise definition at any time of who is doing what with whom, and how well.
- Integrate operational activities and strategic goal with the organization model.
- Perform proactive performance monitoring and control in order to automatically derive reorganization strategy.

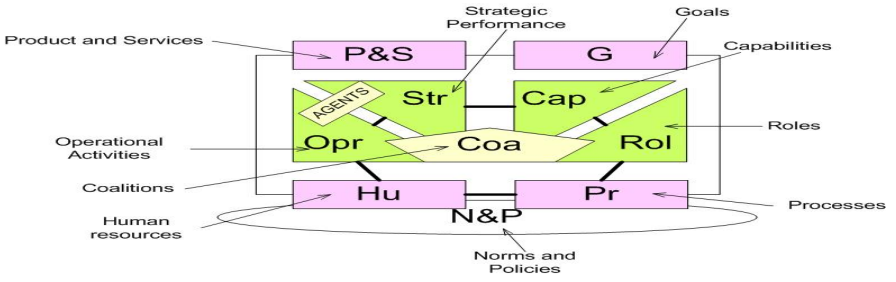


Fig. 1. An integrated organizational model

3 A Generic Approach for Modeling Dynamic Organization

In this section we present the definition of the key elements of our organizational model, along with the coalition formation and its dynamics.

3.1 Basic Concepts

3.1.1 Capabilities

The notion of capability is usually related to both the roles and the agents. A role requires a set of capabilities to accomplish a specific goal, while an agent needs to possess specific capabilities in order to be able to play the role [3]. From a role perspective, we have identified two distinct types of capabilities: *operational* capabilities which only deal with the processing of the activity's tasks, that is its execution, its related communications, and decision makings towards accomplishment of the goal. S*trategic* capabilities, on the other hand cope with the performance management such as the delivery time, the product or service quality, cost, effort etc. Basically the formal looks after of the efficiency aspect of the goal accomplishment, whereas the later considers the effectiveness aspect. As for an agent, it possesses capabilities which are intrinsic to a particular role; this includes data access, data manipulation, and computation.

3.1.2 Roles

We define a role as "a collection of duties and rights" [1]. Duties represent the tasks and interactions that the role is obligated to perform, whereas rights represent the permissions to utilize information entities to perform tasks or interactions. Thus, the concept of role in our system is essentially an abstraction on one hand, for the tasks that are necessary to be performed and/or the interactions that need to occur with other roles to achieve an individual goal, and on the other hand, the information that needs to be accessed or will be generated during the course of performance of those tasks/interactions. We have defined two categories of roles: an *organizational* role and a *conceptual* role.

3.1.2.1 An Organizational Role. An organizational role is related to a physical human agent with clear responsibilities. It may consist of one or more roles depending

on the profile. An organizational role in our model is played by one human agent surrounded by a set of software agents that assist him/her in the operational and/or performance management. This organizational role is represented by a human proxy agent. Figure 2a delineates the two types of capabilities of an organizational role. These are operational and strategic capabilities; the arrow in the middle indicates the possible shift between the two; that is when some of the capabilities are delegated to software agents. The abstraction of an organizational role is presented in Figure 2b and requires a coalition of agents with the different required capabilities.

3.1.2.2 A Conceptual Role. A conceptual role necessitates coalitions of agents of more than one organizational role, and often involves relationships (i.e., collaboration, conflict resolution, etc) among the roles. It requires a set of capabilities necessary to perform an activity. An activity consists of a well defined set of tasks. A conceptual role coalition is founded on the notion of shared activity that associates a goal with a set of agents and a set of tasks.

Both organizational and conceptual role models are task centered ones. A task represents anything which has an agent involvement, attributes, properties and agent means of interactions. A task has both functional and behavioral perspective. The functional perspective describes the hierarchical structure of tasks. Whereas, the behavioral perspective specifies how the tasks should be performed. A task is described by its inputs, outputs, and methods. Inputs are data needed for the execution of the task, outputs are data generated by the task during its execution, and methods are mechanisms describing the execution of the tasks.

3.1.3 Agents

Agents refer to physical human or software agents that implement their respective roles. We defined two types of agents; the *operational agents* responsible for the functional capabilities of the identified roles in the organization, and the *strategic agents* which take care of the performance indicators capabilities. The operational agents receive information from the role in terms of well defined tasks. From this information, operational agents extract performance data (i.e. schedule, deliverable, effort, cost, quality etc...). This performance data is communicated to selected strategic agents. For example, the strategic agent dealing with delivery time receives all the time data for task start, progress, delivery, etc. Operational agents may exchange information with other operational agents, when the task is shared. For the performance indicators, we identified several types of knowledge, which we collected and coded following these steps. First, the list of performance indicators and their relationships was identified, second we incorporate the humans and the organization's preferences in the performance, i.e. which indicators are considered to be most important, which of their values should be considered successful performance and where the individuals/organization should strive for improvement. Identifying these elements is not trivial and requires input from organization's management and domain experts. In coding the knowledge into the software system, we first use interviews to collect this data [4], then construct a comprehensive ontology using protégé tool [9].

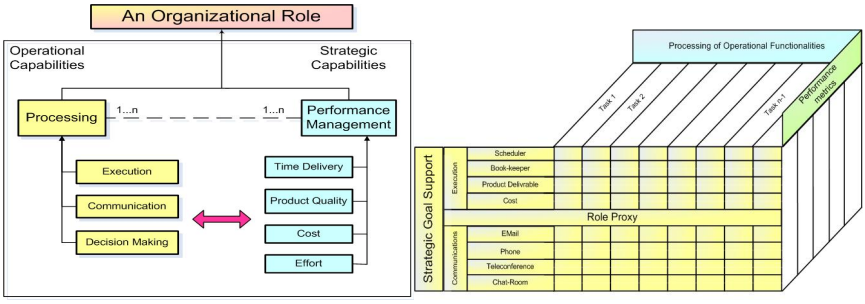


Fig. 2. (a) An organizational role capabilities (b) An abstraction of an organizational role

3.1.4 Coalition

Coalition formation deals with how organizational role (physical human or software agent) plays its role, it consists of binding between an agent and its tasks. It examines the agent's engagement in the coordination process, what makes them keep/leave the coalition and which partners in an organization will work together for a task or how the operational structure will be restructured during the project or organization lifecycle. In the next section we present the formal model of this coalition.

3.2 Formalizing the Coalitions

In this section we present algorithms for coalition formations and their dynamics for both organizational and conceptual roles.

3.2.1 Constructing the Organizational Role Coalition

An organizational role coalition consists of two steps, first the binding between the operational agent(s) and the allocated tasks, second allowing the interaction with other organizational roles when there is task dependencies or organizational hierarchy responsibility (i.e. with the team leader). Table 1 shows the types of communication we identified; (i) we allow the human proxy agent as the only agent to talk to human, (ii) the strategic agents of an organizational role to collaborate among themselves, and (iii) we let one human proxy agent to communicate with another human proxy agent for shared tasks or responsibilities.

For our goal of modeling adaptive organization with human in the loops, the system should capture the operational and strategic capabilities. Here we assume that operational capabilities are carried by human physical agent (H_a), however he/she can delegate those routine tasks to software agents like the work of [2]. Strategic capabilities on the other hand deal with the execution and communication aspect towards the goal. In this paper we dealt only with the execution aspect that is scheduling, bookkeeping, product deliverable, effort, cost, etc, future extension of this work will include the communication part. For the problem addressed, we pursue an example of a software engineering organizational model setting; a member of an organization can be involved in specific project(s) as well as organizational tasks,

these are P_{Ta}, O_{Ta} respectively. Team's project and team's organization tasks are denoted by P_{TT} and O_{TT} correspondingly. The members of the team are T_H and task's dependency function is defined by tdf. Each individual is assisted by five agents: the proxy, coordinator, scheduler, bookkeeper, and product deliverables, (H_P;R_C;R_S;R_B;R_D) respectively (step 2). These agents are assigned the individual's organization and project tasks, and their dependencies (steps3). For every task (project or organization) we allow the cooperation among the supporting agents of an organizational role according to the interaction types identified in table 1. (step5-6). If the task has no dependencies we add the collaboration with the team leader proxy (step 7). If the task has dependencies (i.e. carried by more than one individual) we allow interaction between proxy agents of the individuals sharing the task as well as between the agents of the teams' leaders (step8). The next level of collaboration, deals with the team leader (step 12-15), where he/she is assisted by a proxy, coordinator, scheduler, and product deliverable support agents (step13). The team's project and organization tasks, their dependencies, and the team members list are assigned to the supporting agents of the team leader (step 14).

Table 1. Agent interaction types

	Between	Definition
1	$H_{P1} \infty H_1$	Interaction between human proxy agent and the physical human it represents
2	$R_{i1} \infty R_{j1}$	Interaction among agents that support the same physical human agent
3	$H_{P1} \infty H_{P2}$	Interaction between one human proxy agent and another human proxy agent

Inputs:

$$P_{Ta} = \sum_{i=1}^{n} t_i \; ; \; O_{Ta} = \sum_{j=1}^{m} t_j \; ; \; P_{TT} = \sum_{q=1}^{Q} t_q \; ; \; O_{TT} = \sum_{l=1}^{L} t_l \; ; \quad T_H = \sum_{H=1}^{K} H_k \; ; \quad tdf \; .$$

Outputs:
-Human's Physical agent (H_a) support agents' collaboration binding
-Humans' agent proxy interaction when there is task dependencies, or hierarchy responsibilities

1. for each human physical agent (H_a)
2. $H_a := (H_{Pa}; R_C; R_S; R_B; R_D)$
3. $(H_{Pa}; R_C; R_S; R_B; R_D) := [P_{Ta}; O_{Ta}; tdf]$
4. end for
5. for All P_{Ta} and O_{Ta}
6. $(H_{Pa} \infty H_a)$ and $(H_{Pa} \infty R_C \infty R_S \infty R_B \infty R_D)$
7. if $\neg \exists tdf\,(t_{i,j})$ then $(H_{Pa} \infty H_{TLPa})$
8. else $(H_a \infty H_k)$ and $(H_{Pa} \infty H_{Pk})$ and $(R_{ai} \infty R_{ki})$ and $(H_{Pa} \infty H_{PTLa})$ and $(H_{PTLa} \infty H_{PTLk})$
9. end if
10. go to 12
11. end for

12. for all P_{TT} and O_{TT}
13. $H_{TL} := (H_{TLP}; R_C; R_S; R_D)$
14. $(H_{TLP}; R_C; R_S; R_D) := [\mathrm{P_{TT}}; O_{\mathrm{TT}}; tdf; T_H]$
15. end for

3.2.2 Constructing the Conceptual Role Coalition

A conceptual role coalition is founded on the notion of shared activity that associates a goal with a set of agents and a set of tasks. According to the following semantics; the individuals involved in the activity have a common goal to achieve through the accomplishment of the tasks of the activity. Performing a task requires a binding to it. A conceptual role coalition dynamically binds a set of tasks and agents from a given activity. This binding is normalized by a set of rules specified bellow and managed by an algorithm illustrated in the next subsection. An agent *a* assigned to activity *V* performs a task *t* from *V*, if and only if, he has a binding on *t*. We call $c_v(A, T)$ a conceptual role coalition of the activity *V* binding a set of agents *A* to a set of tasks *T* from *V*, such that every agent of *A* is assigned to a task from *T*. The conceptual role coalition combines the collaborations of the agents with the tasks. A given coalition shows who is sharing tasks with whom, and which tasks are shared. The conceptual role coalition is based on the following rules.

Rule 1: The set *A* of agents of the conceptual role is maximum that is *A* contains all the members of the activity bound on any of the tasks of the activity.
Rule 2: The set *T* of tasks of the conceptual role is maximum that is *T* contains all the tasks of the activity that the members of the conceptual role are sharing (a task is shared if and only if, more than one binding exist).

To ensure rules 1 and 2, we define the following operation.

Operation: Two conceptual roles can be merged if and only if they have the same agent members or they have the same tasks.

The result of merging two conceptual role coalition $c_{1V}(A;T)$ and $c_{2V}(A';T')$ is an other conceptual role $c_{3V}(A \cup A'; T \cup T')$.

For example $c_{1V}(\{a_1,a_2\};\{t_1,t_2,t_3\})$ is mergeable with $c_{2V}(\{a_1,a_2\};\{t_4\})$ and the result is $c_{3V}(\{a_1,a_2\};\{t_1,t_2,t_3,t_4\})$. Similarly $c_{1V}(\{a_1,a_2\};\{t_1\})$ and $c_{2V}(\{a_3,a_4\};\{t_1\})$ are mergeable in $c_{3V}(\{a_1,a_2,a_3,a_4\}; \{t_1\})$.

The binding of agents and tasks control the evolution of the coalition set of an activity but must guarantee rules 1 and 2.

3.2.2.1 Illustrative Example. We demonstrate the coalition dynamic methods through an example; in this example an activity V has four agent members: a_1, a_2, a_3, and a_4 that are responsible for t_1, t_2, t_3, t_4 and t_5 in this way:

- a_1, a_2 and a_3 are collaborating on the tasks t_1, t_2, and t_3.
- a_1 and a_2 are collaborating on t_4.
- a_4 is active on t_5.

This is represented by three conceptual role coalitions

$c_{1V}(\{a_1,a_2,a_3\};\{t_1,t_2,t_3\})$;$c_{2V}(\{a_1,a_2\};\{t_4\})$ and $c_{3V}(\{a_4\};\{t_5\})$(see Figure 3(a)).

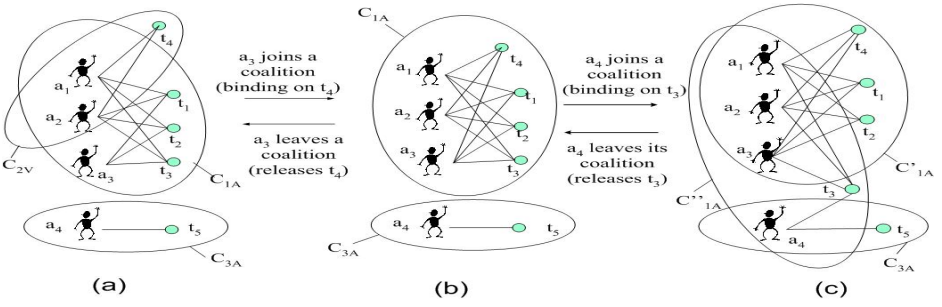


Fig. 3. Conceptual role coalition dynamics

3.2.2.2 Coalition Dynamics

Upon Joining the Coalition

If the agent a_3 will be involved in task t_4, he has to bind to t_4, then the conceptual role coalition involving t_4 must show that a_3 joins the group of agents responsible for t_4, so a_3 will be added to c_{2V} which becomes $c_{2V}(\{a_1,a_2,a_3\};\{t_4\})$. However this will break the rule 2, because conceptual role coalitions c_{1V} and c_{2V} have the same members. The two conceptual role coalitions have to be merged; and the coalition set becomes: $c_{1V}(\{a_1,a_2,a_3\};\{t_1, t_2, t_3, t_4\})$; and $c_{3V}(\{a_4\};\{t_5\})$(see Figure 3(b)).

If the agent a_4 becomes involved on t_3, then a_4 has to be added to coalition c_{1V}, but this will violate rule 1 because a_4 is not bound to t_1, t_2, t_4. Therefore the coalition c_{1V} will be split into $c'_{1V}(\{a_1, a_2, a_3\};\{t_1, t_2, t_4\})$ and $c''_{1V}(\{a_1,a_2, a_3 , a_4\};\{t_3\})$. These two new conceptual role coalitions are not mergeable with any of the other coalitions of the activity, so the resulting coalition set is: $c'_{1V}(\{a_1,a_2, a_3\};\{t_1, t_2, t_4\})$ and $c''_{1V}(\{a_1,a_2,a_3,a_4\};\{t_3\})$, $c_{3V}(\{a_4\};\{t_5\})$.(see Figure 4(c)).

We construct the conceptual role join coalition algorithm as fellow:

Let (m,t) be the new bindings, in order to ensure rules 1 and 2, we present the algorithm in two steps. The first step consist of finding the conceptual role coalition c_V involving task t, if such coalition exists, and then adding the member m to c_V member set. The second step is the merge of the coalitions.

We arrange the coalitions in three sets:

- ToChange: includes the coalition that will be modified.
- Mergeable: contains the possible mergeable coalitions: it is composed of the remaining coalitions having agent member m as a member.
- UnChanged: contains the coalitions that will be left unmodified, it is composed of the remaining coalitions.

Upon Leaving the Coalition

When the agent leave the coalition and release the binding on task t, this must be reflected in the relevant conceptual role coalition. The first step is to find the relevant coalition that is finding the coalition involving the task t. The second step is to remove the agent from the list of the agent of this conceptual role coalition, and to guarantee that the resulting coalition does still correspond to the coalition definition and still respects rules 1 and 2.

This restriction will imply a reorganization of the conceptual role coalition. Using the previous example, we had the following coalition set:
{c'$_{1V}$({a$_1$,a$_2$,a$_3$};{t$_1$,t$_2$,t$_4$}), c''$_{1V}$({a$_1$,a$_2$, a$_3$, a$_4$};{t$_3$}), c$_{3A}$({ a$_4$};{t$_5$})} (See Figure 4(c)).

If agent a_3 releases the task t_4, it has to be removed from the member associated with t_4 in c'_{1V}. Rule 2 implies splitting the coalition into c_{1V} *({a$_1$,a$_2$, a$_3$};{t$_1$, t$_2$})* and c_{2V}*({a$_1$, a$_2$};{t$_4$})*

Algorithm
We consider the following:

- $c_V(\zeta;T)=\zeta$ with T a set of tasks.
- $c_V(A;\zeta)=\zeta$ with A a set of agents member of the activity V.
- Given C_V the set of the coalitions of an activity V, a_1 an agent and t_1 a task of V.

```
//--------- Upon joining the coalition
Begin
consider
     ToChange = {c_V(A;T), c_V ∈ C_V, t_1 ∈ T}
     Mergeable = {c'_V(A;T'), c_V ∈ C_V, a_1 ∈ A}
     UnChanged = C_V -ToChange-Mergeable
if ToChange = ζ then
          if ∃c'_V({a_1});T') ∈ Mergeable then
             C_V := C_V - c'_v ∪ {c'_V({a_1};T' ∪ {t_1})}
          else
            form c"_V({a_1};{t_1})
            C_V := C_V ∪ {c"_V}
else
     consider ToChange = {c_V(A;T)} (with t_1 ∈ T)
     if ∃c'_A(A ∪ {a_1};T') ∈ Mergeable then
          C_V := C_V - c_V - c'_V ∪ {c_V(A;T - {t_1}), c'_V(A ∪ {a_1};T' ∪ {t_1})}
     else
          C_V := C_V - c_V ∪ {(A;T - {t_1}), c'_V(A ∪ {a_1};{t_1})}
end
//------- Upon leaving the coalition
Begin
consider
     ToChange = {c_V(A ∪ {a_1};T ∪ {t_1})}
     Mergeable = {c'_V(A;T'), c_V ∈ C_V, a_1 ∉ A}
     UnChanged = C_V -ToChange-Mergeable
     if A = ζ then
```

if $T = \zeta$ and Unchanged $\neq \zeta$ then

$C_V := C_V - c_V$

else $C_V := C_V - c_V \cup \{c'_V(\{a_1\};\zeta)\}$

else

if $T = \zeta$ then

if Unchanged $= \zeta$ then

$C_V := C_V - c_V \cup \{c'_V(A;\{t_1\}), c''_V(\{a_1\};\zeta)\}$

else $C_V := C_V - c_V \cup \{c'_V(A;\{t_1\})\}$

else

if $\exists c'''_V(A;T''') \in$ Mergeable then

$C_V := C_V - c_V - c_V''' \cup \{c'_V(A \cup \{a_1\};T), c''_V(A;T''' \cup \{t_1\})\}$

else $C_V := C_V - c_V \cup \{c'_V(A \cup \{a_1\};T), c''_V(A;\{t_1\})\}$

end

4 System Architecture

We sketch the basic components of our overall system architecture (cf. figure 4). It consists of three main modules: (1) the users' platform, (2) the shared organizational data, norms, processes and projects data, (3) and an intelligent agent-based control module. The prototype provides the ability to trace the work done during both autonomy periods and collaborative periods. By storing main states of all activities, duration of these activities, missed deadlines or decisions taken, we provide a view of what happened and when and where the process failed or succeeded. As illustrated determining the activity's tasks evolution and their performance is highly relevant for organizational competencies. The incorporation of this kind of knowledge required an integrated monitoring of the concept of capabilities, roles, agents and coalitions that abstract the human in the loop as well as the process. The intelligent control module contains mechanisms for coalition management presented in the previous section, and allows adaptability during lifecycle. The interface of module 3 displays the information on activities evolution and agents' involvement, as well as their performance at the current time. This assists organizations in identifying potential problems in order to take corrective actions as required. The shared data module stores organization's policies and norms, projects' processes, human agent profiles, organization assignment of activities, tasks and roles. At the user side, we design a set of collaborative strategic agents (scheduler, bookkeeper, product deliverable, etc,) that collect the data from wrapper tools used to execute the tasks, as well as inputs from human users. The current implementation is based on a distributed architecture that uses java and Jade tool [5]. Each member of the organization team hosts a part of the organization and project data repository and exchanges are done using SOAP [12] messages.

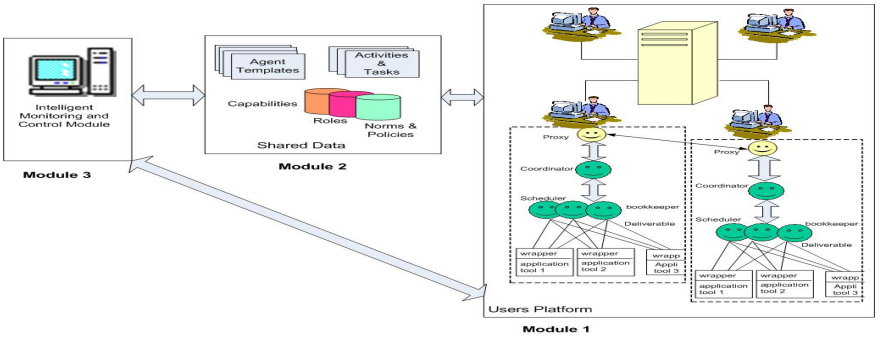


Fig. 4. System architecture

5 Related Works

There are essentially three contribution made by this paper: first, the idea of including human in the loop for organizational modeling; second, the combination of operational capabilities and strategic performance management when assisting organization teams; and third the notion of capturing the dynamic organizational knowledge which is often presented in an implicit manner. We will situate out our work in relation to each of the categories.

Authors of [14], [3] and [7] propose organizational modeling using agents; they share the concept of role, agent in their modeling approaches. Although, the work of [3] describes an adaptable organization modeling, the human in the loop was not explicitly captured in this model. The work of [14] presents a theoretical methodology for role based modeling where agents take and release roles at run time, however it was not verified in real world applications and it also does not take advantage of the power of including human in the modeling loop. Authors in [7] propose a modeling approach to build an agent based information system targeting interaction between operational and strategic goals; however their model does not clarify how the two capabilities are integrated.

With regards using agents to support individuals and teams, work of [13], [6], and [2] tackled this aspect, however their approaches targeted some specific applications with no formal foundation , their generality was not verified, and there was not a clarity between the operational and strategic capabilities, physical human agents also were not explicitly modeled as well as their dynamics. Authors of [10] propose a data-driven process control and exception handling for the logistic area, the work focuses only on handling exception with regards process data, however exception can include agent joining and/or leaving the environment, the notion of organizational and conceptual role coalitions presented in this work makes more general strides.

6 Conclusion and Future Works

We have presented a generic approach for modeling adaptive organizations with human agents in the loop. This approach is based on four pillars: capabilities, roles, agents, and coalitions. It integrates operational and strategic capabilities with mechanisms for different types of coalitions; this allows organizations to handle their

resources and processes in a very elegant and user friendly manner. Ongoing work deals with simulation of the model using the third generation distributed dynamic decision making DDD-III [11] for a software engineering project case study involving different performance indicators i.e. delivery time, product deliverable quality, effort, cost, etc.

Acknowledgments

The authors are grateful to IBM for the 2006 UIMA Innovation Award and grants from the NSERC that were used to partially support this project, and would like to acknowledge the clarifications and suggestions made by Dr. Joerg Denzinger on early versions of this work. We would also like to thanks Dr Kevin M. Gildea and Dr. Georgiy Levchuk of Aptima, Inc. for providing us with the DDD-III simulator to validate the model.

References

1. Biddle, B.J., Thomas, E.J.: Role Theory: Concepts and Research. John Wiley & Son, Inc. West Sussex, England (1966)
2. Chalupsky, H., Gil, Y., Knoblock, C.A., Lerman, K., Oh, J., Pynadath, D.V., Russ, T.A., Tambe, M.: Electric Elves: Applying Agent Technology to Support Human Organizations. In: IAAI 2001, pp. 51–58 (2001)
3. DeLoach, S.A., Matson, E.: An Organizational Model for Designing Adaptive Multiagent Systems. In: The AAAI-04 Workshop on Agent Organizations: Theory and Practice (AOTP) (2004)
4. Houari, N., Far, B.H.: Intelligent Agents Based Architecture for Distributed Decision Support to Manage a Project's Lifecycle (Ethics Review Report), University of Calgary CE101-3751 (2005)
5. JADE, Home Page: http://jade.tilab.com/
6. Kogut, P.A., Yen, J., Leung, Y., Sun, S., Wang, R., Mielczarek, T., Hellar, B.: Proactive Information Gathering For Homeland Security Teams. Commun. ACM 47(3), 48–50 (2004)
7. Krauth, E., Moonen, H., Popova, V., Schut, M.: Agent-Based Modeling of Third-Party Logistics Performance Indicators. In: Intelligent Agent Technology IEEE/WIC/ACM International (IAT 2005), pp. 262–265 (2005)
8. Meirina, C., Ruan, S., Kleinman, D.L.: Real-time agent-based decision support system to facilitate effective organizational adaptation. In: IEEE Systems, man, and cybernetics society, vol. 3, pp. 2681–2686. IEEE Computer Society Press, Los Alamitos, USA (2004)
9. Protégé, Home Page: http://protege.stanford.edu/
10. Rinderle, S., Reichert, M.: Data-Driven Process Control and Exception Handling in Process Management Systems. In: Dubois, E., Pohl, K. (eds.) CAiSE 2006. LNCS, vol. 4001, pp. 273–287. Springer, Heidelberg (2006)
11. Simulator: The DDD-III: Distributed Dynamic Decision-making Simulator Provided by Aptima, Inc. (2006)
12. SOAP, Home Page: http://www.w3.org/2000/xp/Group/
13. Sycara, K.P., Paolucci, M., Velsen, M.V., Giampapa, J.A.: The RETSINA MAS Infrastructure. Autonomous Agents and Multi-Agent Systems 7(1-2), 29–48 (2003)
14. Xu, H., Zhang, X.: A Methodology for Role-Based Modeling of Open Multi-Agent Software Systems. In: Xu, H., Zhang, X. (eds.) 7th International Conference on Enterprise Information Systems (ICEIS 3, 2005), pp. 246–253 (2005)

Multidimensional Fuzzy Integrals

Yasuo Narukawa[1] and Vicenç Torra[2]

[1] Toho Gakuen,
3-1-10 Naka, Kunitachi, Tokyo, 186-0004 Japan
narukawa@d4.dion.ne.jp
[2] Institut d'Investigació en Intel·ligència Artificial,
Campus de Bellaterra, 08193 Bellaterra, Catalonia, Spain
vtorra@iiia.csic.es

Abstract. This paper studies multidimensional fuzzy integrals. We consider a multidimensional generalized fuzzy integral that generalizes the multidimensional Choquet and Sugeno integrals. Then, an extension of part of Fubini's theorem is given. Finally, citation analysis is considered as an example of the application of the results presented here.

Keywords: fuzzy integral, Choquet integral, Sugeno integral, multidimensional integrals.

1 Introduction

In a recent paper [10] we studied fuzzy multidimensional integrals motivated by their application on citation analysis. We showed that when computing indices for evaluating research, it is meaningful to aggregate the citations of a researcher with respect to the publications and the years in which these citations are found. The standard approach of counting citations as well as the Hirsch index [4] can be seen from this perspective [12].

In this paper we present new mathematical results that can be applied in this setting. Formally, we consider multidimensional generalized fuzzy integrals. We introduce them and prove that under some conditions the order of integration is not relevant. This result generalizes the one by Machida in [8].

Our approach for integrating multidimensional functions is only valid for a restricted type of functions. In fact, it is usual that functions representing the number of citations do not belong to such type of functions. To permit arbitrary functions we introduce an extension of the domain and consider the integrals in this new setting. A similar theorem is proven in this alternative framework.

The structure of this paper is as follows. First, we review in Section 2 some concepts that are needed later on in this paper. Then, in Section 3 we will consider the generalized fuzzy integral, and in Section 3 multidimensional integrals. Section 5 considers the application of multidimensional integrals for citation analysis. The paper finishes with some conclusions.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 145–155, 2007.

2 Preliminaries

Next, we present the basic definitions of a fuzzy measure.

Definition 1. *Let X be a universal set and $\mathcal{X}$ be a subset of 2^X. $(X, \mathcal{X})$ is called a fuzzy measurable space. We say that a function $f : X \to \mathbb{R}^+$ is $\mathcal{X}$-measurable if $\{x|f(x) \geq a\} \in \mathcal{X}$ for any $a \geq 0$.*

Definition 2. [3] Let f and g are $\mathcal{X}$-measurable functions on X. We say that f and g are comonotonic if

$$f(x) < f(y) \Rightarrow g(x) \leq g(y)$$

for $x, y \in X$.

Definition 3. [11] Let $(X, \mathcal{X})$ be a fuzzy measurable space. A fuzzy measure μ on $(X, \mathcal{X})$ is a real valued set function, $\mu : \mathcal{X} \longrightarrow \mathbb{R}^+$ with the following properties. A triplet $(X, \mathcal{X}, \mu)$ is said to be a fuzzy measure space.

(1) $\mu(\emptyset) = 0$, $\mu(X) = 1$
(2) $\mu(A) \leq \mu(B)$ whenever $A \subset B$, $A, B \in \mathcal{X}$.

Definition 4. [2,9] Let $(X, \mathcal{X}, \mu)$ be a fuzzy measure space and f be a $\mathcal{X}$-measurable function.

The Choquet integral of f with respect to μ is defined by

$$(C)\int f d\mu := \int_0^\infty \mu_f(r)dr,$$

where $\mu_f(r) = \mu(\{x|f(x) \geq r\})$.

If a $\mathcal{X}$-measurable function f is a simple function, that is, $f(x) := \sum_{i=1}^{n} a_i 1_{A_i}$ $a_i \geq 0$ $A_1 \supset A_2 \supset \ldots A_n$,$A_i \in \mathcal{X}$ we have

$$(C)\int f d\mu = \sum_{i=1}^{n} a_i \mu(A_i).$$

Definition 5. [11] Let $(X, \mathcal{X}, \mu)$ be a fuzzy measure space and $f : X \to [0, 1]$ be a $\mathcal{X}$-measurable function. The Sugeno integral of f with respect to μ is defined by

$$(S)\int f d\mu := \sup_{r \in [0,1]} [r \wedge \mu_f(r)]$$

where $\mu_f(r) := \mu(\{x|f(x) \geq r\})$.

If f is a simple function, the Sugeno integral is written as

$$(S)\int f d\mu = \bigvee_{i=1}^{n} ((a_1 + \cdots + a_i) \wedge \mu(A_i)).$$

3 Generalized Fuzzy Integral

In this section, we use $t-$conorms and $t-$norms. They are binary operators that generalize addition and multiplication, and also max and min. In the following, we assume that the universal set X is a finite set.

Definition 6. *A triangular norm (t-norm)* $\top$ *is a binary operation on* $[0,1]$ *fulfilling the conditions:*

(T1) $x\top 1 = 1$.
(T2) $x\top y \leq u\top v$ *whenever* $x \leq u$ *and* $y \leq v$.
(T3) $x\top y = y\top x$.
(T4) $(x\top y)\top z = x\top(y\top z)$.

A triangular conorm (t-conorm) $\perp$ *is a binary operation on* $[0,1]$ *fulfilling the conditions:*

(T1) $x\perp 0 = x$.
(T2) $x\perp y \leq u\perp v$ *whenever* $x \leq u$ *and* $y \leq v$.
(T3) $x\perp y = y\perp x$.
(T4) $(x\perp y)\perp z = x\perp(y\perp z)$.

A t-conorm is said to be strict if and only if it is continuous on $[0,1]$ *and strictly increasing in each of its places. A t-conorm* $\perp$ *is said to be Archimedean if and only if* $x\perp x > x$ *for all* $x \in (0,1)$.

Example 1

(1) The maximum operator $x \vee y$ is a non Archimedean t-conorm.
(2) The bounded sum $x\hat{+}y := 1 \wedge (x+y)$ is an Archimedean t-conorm.
(3) The Sugeno operator $x +_\lambda y := 1 \wedge (x + y + \lambda xy)$ $(-1 < \lambda < \infty)$ is an Archimedean t-conorm.

Proposition 1. [7] *If a t-conorm* $\perp$ *is continuous and Archimedean, then there exists a continuous and strictly increasing function* $g : [0,1] \to [0,\infty]$ *such that* $x\perp y = g^{(-1)}(g(x)+g(y))$, *where* $g^{(-1)}$ *is the pseudo inverse of* g *which is defined by*

$$g^{(-1)}(u) := \begin{cases} g^{(-1)}(u) & \text{if } u \leq g(1) \\ 1 & \text{if } u > g(1). \end{cases}$$

The function g *is called an additive generator of a t-conorm* $\perp$.

Example 2. Let P be a probability measure on $(X, 2^X)$. Define a function $\varphi_\lambda : R \to R$ by

$$\varphi_\lambda(r) := \log_{1+\lambda}(1+\lambda r)$$

and a fuzzy measure μ_λ by $\mu_\lambda := \varphi_\lambda^{-1} \circ P$.

φ_λ is the additive generator of the Sugeno operator $+_\lambda$.

Definition 7. *Let m be a fuzzy measure on a fuzzy measurable space $(X, \mathcal{X})$. We say that m is $\perp$-decomposable if $m(A \cup B) = m(A) \perp m(B)$ whenever $A \cap B \neq \emptyset$ for $A, B \in \mathcal{X}$. A $\perp$-decomposable fuzzy measure m is called normal if $\perp = \vee$ or $g \circ m$ is an infinite additive measure or $g \circ m$ is a finite additive measure, where g is a generator of $\perp$.*

Definition 8. Let $\perp$ be a continuous t-conorm, which is $\vee$ or Archimedean. Define a non decreasing operator $\boxdot : [0,1] \times [0,1] \to [0,1]$ satisfying

(M1) $\boxdot$ is left continuous t-norm on $(0,1]$.

(M2) $a \boxdot x = 0$ if and only if $a = 0$ or $x = 0$.

(M3) if $x \perp y < 1$ then $a \boxdot (x \perp y) = (a \boxdot x) \perp (a \boxdot y)$.

(M4) if $a \perp b < 1$ then $(a \perp b) \boxdot x = (a \boxdot x) \perp (b \boxdot x)$.

We say that $(\perp, \boxdot)$ is a t-system.

For a given t-conorm $\perp$, we define an operation $-_\perp$ by

$$a -_\perp b := \inf\{c | b \perp c \geq a\}$$

for all $(a, b) \in [0,1]^2$. We say that $-_\perp$ is a pseudo inverse of $\perp$.

Definition 9. Let m be a fuzzy measure on a fuzzy measurable space $(X, \mathcal{X})$, and $(\perp, \boxdot)$ be a t-system. If m is a normal $\perp-$ decomposable fuzzy measure, then for a function $f : X \to [0,1]$ such that $f = \perp_{i=1}^{n} r_i 1_{D_i}$ where $D_i \cap D_j \neq \emptyset$ for $i \neq j$, the t-conorm integral is defined as follows:

$$(T) \int f \boxdot dm := \perp_{i=1}^{n} r_i \boxdot m(D_i).$$

Example 3. A uninorm [13] is a binary operation on the unit interval which is commutative, associative, non-decreasing in each component, and which has a neutral element. Let $([0,1], S, U)$ be a conditionally distributive semiring [5]; that is, S is a continuous $t-$conorm and U is a left continuous uninorm satisfying the conditional distributivity of U over S: for all $x, y, z \in [0,1]$ with $S(x,y) < 1$ we have $U(x, S(y,z)) = S(U(x,y), U(x,z))$. Suppose that $U(0,x) = 0$ for $x \in [0,1]$. In the case of $\perp = S$ and $\boxdot = U$, the $t-$ conorm integral in Definition 9 coincides with $(S,U)-$ integral by Klement, Mesiar and Pap [6].

Definition 10. Let m be a fuzzy measure on a measurable space $(X, \mathcal{X})$, and $(\perp, \boxdot)$ be a t-system. For a measurable function $f(x) := \sum_{i=1}^{n} a_i 1_{A_i}$ where $a_i \geq 0$ $A_1 \supset A_2 \supset \ldots A_n$,$A_i \in \mathcal{X}$, the generalized t-conorm integral (GT-integral) is defined as follows:

$$(GT) \int f \boxdot dm := \perp_{i=1}^{n} a_i \boxdot m(A_i).$$

The next proposition follows from the definition of the pseudo inverse $-_{\perp}$, the generalized t-conorm integral (Definition 10) and the t-conorm integral (Definition 9).

Proposition 1. *Let m be a fuzzy measure on a measurable space $(X, \mathcal{X})$, and $(\perp, \boxdot)$ be a t-system. If m is a normal $\perp$-decomposable fuzzy measure, the generalized t-conorm integral coincides with the t-conorm integral.*

Example 4. (1) If $\perp = \hat{+}$ and $\boxdot = \cdot$, the generalized t-conorm integral is a Choquet integral.
(2) If $\perp = \vee$ and $\boxdot = \wedge$, the generalized t-conorm integral is a Sugeno integral.

Let measurable functions f, g be comonotonic. Since we have $\{x|f(x) \geq a\} \subset \{x|g(x) \geq b\}$ or $\{x|f(x) \geq a\} \supset \{x|g(x) \geq b\}$ for every $a, b \in [0, 1]$, we have

$$(f \perp g)(x) := \sum_{i=1}^{n} a_i 1_{A_i}$$

where $a_i \geq 0$ $A_1 \supset A_2 \supset \ldots A_n$,$A_i \in \mathcal{X}$, Therefore we have the next theorem

Theorem 1. *Let $(X, \mathcal{X}, \mu)$ be a fuzzy measure space and $(\perp, \boxdot)$ be a t-system. If measurable functions f, g are comonotonic; then, we have*

$$(GT) \int (f \perp g) \boxdot dm = (GT) \int f \boxdot dm \perp (GT) \int g \boxdot dm.$$

We say that the property above is the $\perp$-additivity of generalized fuzzy integral.

4 Multidimensional Integrals

4.1 Measurable Function

We consider below the case of the product of two fuzzy measurable spaces. Let X and Y be two universal sets and $X \times Y$ be a direct product of X and Y. Let $(X, \mathcal{X})$ and $(Y, \mathcal{Y})$ be fuzzy measurable spaces. Now, let us define the following class of sets:

$$\mathcal{X} \times \mathcal{Y} := \{A \times B | A \in \mathcal{X}, B \in \mathcal{Y}\}$$

We will consider a measurable space $(X \times Y, \mathcal{X} \times \mathcal{Y})$.

Suppose that $\mathcal{X} := 2^X$ and $\mathcal{Y} := 2^Y$. Note that $\mathcal{X} \times \mathcal{Y} \neq 2^{X \times Y}$. Therefore the class of $\mathcal{X} \times \mathcal{Y}$-mesurable function is smaller than $2^{X \times Y} - measurable function$.

Example 5. *Let $X := \{x_1, x_2\}$ and $Y := \{y_1, y_2\}$. We have*

$$\begin{aligned} 2^X \times 2^Y := \{&\{(x_1, y_1)\}, \{(x_1, y_2)\}, \{(x_2, y_1)\}, \\ &\{(x_2, y_2)\}, \{(x_1, y_1), (x_2, y_1)\}, \{(x_1, y_2), (x_2, y_2)\}, \\ &\{(x_1, y_1), (x_1, y_2)\}, \{(x_2, y_1), (x_2, y_2)\}, \\ &\{(x_1, y_1), (x_1, y_2), (x_2, y_2), (x_2, y_2)\}\} \end{aligned}$$

Hence $\{(x_1, y_1), (x_2, y_2)\} \notin 2^X \times 2^Y$.

Let $f : X \times Y \to [0,1]$ be a $\mathcal{X} \times \mathcal{Y}$-mesurable function. Then, for every $a \geq 0$ there exist $A \in \mathcal{X}$ and $B \in \mathcal{Y}$ such that $A \times B = \{(x,y)|f(x,y) \geq a\}$. Then we have the next proposition immediately.

Proposition 2. *Let $f : X \times Y \to [0,1]$ be a $\mathcal{X} \times \mathcal{Y}$-mesurable function.*

(1) *For fixed $y \in Y$, $f(\cdot, y)$ is $\mathcal{X}$-measurable.*
(2) *For fixed $x \in X$, $f(x, \cdot)$ is $\mathcal{Y}$-measurable.*

Example 6. *Let $X := \{x_1, x_2\}$ and $Y := \{y_1, y_2\}$. Consider two fuzzy measurable space $(X, \mathcal{X})$ and $(Y, \mathcal{Y})$ where $\mathcal{X} = 2^X$ and $\mathcal{Y} = 2^Y$.*

(1) *Define a function $f : X \times Y \to [0,1]$ by*
$f(x_1, y_1) = f(x_1, y_2) = 0.2$,
$f(x_2, y_1) = 0.6, f(x_2, y_2) = 1$.
Then we have

$$\begin{aligned}\{(x,y)|f(x,y) \geq 1\} &= \{(x_2, y_2)\} = \{x_2\} \times \{y_2\},\\ \{(x,y)|f(x,y) \geq 0.6\} &= \{(x_2, y_1), (x_2, y_2)\} = \{x_2\} \times \{x_1, y_2\},\\ \{(x,y)|f(x,y) \geq 0.2\} &= \{(x_1, y_1), (x_1, y_2), (x_2, y_1), (x_2, y_2)\}\\ &= \{x_1, x_2\} \times \{y_1, y_2\}.\end{aligned}$$

Therefore f is $\mathcal{X} \times \mathcal{Y}$-measurable.

(2) *Define a function $g : X \times Y \to [0,1]$ by*
$g(x_1, y_1) = 0.2$, $g(x_1, y_2) = 0.4$,
$g(x_2, y_1) = 0.6$, $g(x_2, y_2) = 1$.
Then we have
$\{(x,y)|g(x,y) \geq 0.4\} = \{(x_1, x_2), (x_2, y_1), (x_2, y_2)\} \notin \mathcal{X} \times \mathcal{Y}$.
Therefore g is not $\mathcal{X} \times \mathcal{Y}$-measurable. In fact, if $A \in \mathcal{X} \times \mathcal{Y}$, then we have $|A| = 0, 1, 2, 4$.

As we have seen the example above, we have the next proposition.

Proposition 3. *Let $(X \times Y, \mathcal{X} \times \mathcal{Y})$ be a measurable space and $f : X \times Y \to [0,1]$. If f is $\mathcal{X} \times \mathcal{Y}$-measurable, then $|\{(x,y)|f(x,y) \geq a\}|$ is a divisor of $|X| \times |Y|$ for all $a \in [0,1]$.*

4.2 Multidimensional Generalized Fuzzy Integral

Let $(X, \mathcal{X}, \mu)$ and $(Y, \mathcal{Y}, \nu)$ be two fuzzy measure spaces, $f : X \times Y \to [0,1]$ be a $\mathcal{X} \times \mathcal{Y}$-mesurable function and $(\perp, \boxdot)$ be a t-system. Then f can be represented as

$$f(x) := \sum_{i=1}^{n} a_i 1_{A_i \times B_i}$$

where $a_i \geq 0$, $A_1 \supset A_2 \supset \ldots A_n$,$A_i \in \mathcal{X}$, $B_1 \supset B_2 \supset \ldots B_n$,$B_i \in \mathcal{Y}$.

Since $1_{A \times B} = 1_A \boxdot 1_B$, we have $f(x) := \sum_{i=1}^{n} a_i 1_{A_i} \times 1_{B_i}$. Then we have

$$(GT) \int f d\nu = \perp_{i=1}^{n} a_i 1_{A_i} \boxdot \nu(B_i) = \perp_{i=1}^{n} a_i \boxdot \nu(B_i) 1_{A_i}.$$

Since each 1_{A_i} is comonotonic for all i, it follows from comonotonic $\perp$-additivity that

$$\begin{aligned}(GT)\int((GT)\int f d\nu)d\mu &= (GT)\int(\perp_{i=1}^{n} a_i \boxdot \nu(B_i)1_{A_i})d\mu \\ &= \perp_{i=1}^{n}(GT)\int(a_i \boxdot \nu(B_i)1_{A_i})d\mu \\ &= \perp_{i=1}^{n} a_i \boxdot \nu(B_i) \boxdot \mu(A_i) \\ &= \perp_{i=1}^{n} a_i \boxdot \mu(A_i) \boxdot \nu(B_i).\end{aligned}$$

On the other hand, since $f(x) := \sum_{i=1}^{n} a_i 1_{B_i} \times 1_{A_i}$, we have

$$(GT)\int f d\mu = \perp_{i=1}^{n} a_i 1_{B_i} \boxdot \mu(A_i) = \perp_{i=1}^{n} a_i \boxdot \mu(A_i)1_{B_i}.$$

Since each 1_{B_i} is comonotonic for all i, it follows from comonotonic $\perp$-additivity that

$$\begin{aligned}(GT)\int((GT)\int f d\mu)d\nu &= (GT)\int(\perp_{i=1}^{n} a_i \boxdot \mu(A_i)1_{B_i})d\mu \\ &= \perp_{i=1}^{n}(GT)\int(a_i \boxdot \mu(A_i)1_{B_i})d\mu \\ &= \perp_{i=1}^{n} a_i \boxdot \mu(A_i) \boxdot \nu(B_i).\end{aligned}$$

Define a fuzzy measure m on $\mathcal{X} \times \mathcal{Y}$ by $m(A \times B) := \mu(A) \times \nu(B)$ for $A \times B \in \mathcal{X} \times \mathcal{Y}$. We have the next theorem, which is the main theorem in this paper.

Theorem 2. *Let $(X, \mathcal{X}, \mu)$ and $(Y, \mathcal{Y}, \nu)$ be fuzzy measure spaces. Suppose that $f : X \times Y \to [0, 1]$ be a $\mathcal{X} \times \mathcal{Y}$-mesurable function. Then there exists a fuzzy measure m on $\mathcal{X} \times \mathcal{Y}$ such that*

$$(GT)\int((GT)\int f d\mu)d\nu = (GT)\int f dm = (GT)\int((GT)\int f d\nu)d\mu.$$

Since GT-integral is the generalization of both Choquet integral and Sugeno integrals, considering $\perp = \hat{+}$, $\boxdot = \cdot$, we have the next corollary which was proven by Machida [8].

Corollary 1. *[8] Let $(X, \mathcal{X}, \mu)$ and $(Y, \mathcal{Y}, \nu)$ be two fuzzy measure spaces and f be a $\mathcal{X} \times \mathcal{Y}$-measurable function.*

Then, there exists a fuzzy measure m on $\mathcal{X} \times \mathcal{Y}$ such that

$$(C)\int((C)\int f(x,y)d\mu)d\nu = \quad (C)\int((C)\int f(x,y)d\nu)d\mu = (C)\int f(x,y)dm$$

Considering the characteristic function of $A \times B \in \mathcal{X} \times \mathcal{Y}$, we have $m = \mu\nu$.

In case of Sugeno integral, let $\perp = \vee, \boxdot = \wedge$ we have the next corollary in the same way.

Corollary 2. *[10] Let $(X, \mathcal{X}, \mu)$ and $(Y, \mathcal{Y}, \nu)$ be two fuzzy measure spaces and f be a $\mathcal{X} \times \mathcal{Y}$-measurable function.*

Then, there exists a fuzzy measure m on $\mathcal{X} \times \mathcal{Y}$ such that

$$(S)\int((S)\int f(x,y)d\mu)d\nu = \quad (S)\int((S)\int f(x,y)d\nu)d\mu = (S)\int f(x,y)dm$$

Considering the characteristic function of $A \times B \in \mathcal{X} \times \mathcal{Y}$, we have $m = \mu \wedge \nu$.

4.3 Extension of Domain

Let $(X, \mathcal{X})$ and $(Y, \mathcal{Y})$ be fuzzy measurable spaces. Suppose that$|X|, |Y|$ are finite and $\mathcal{X} := 2^X$ and $\mathcal{Y} := 2^Y$. As we have shown in Section 4.1, the class of $\mathcal{X} \times \mathcal{Y}$ is smaller than $2^{X\times Y}$. Therefore the class of this measurable functions is small. If there is no other assumption on the fuzzy measure, it is impossible to extend the domain. If we assume that μ on $(X, \mathcal{X})$ and ν on $(Y, \mathcal{Y})$ are normal $\perp$-decomposable fuzzy measures; then, it is possible to extend the domain.

Define a class $\overline{\mathcal{X} \times \mathcal{Y}}$ of set $A \in 2^{X\times Y}$ by

$$\overline{\mathcal{X} \times \mathcal{Y}} := \{A \in 2^{X\times Y} | A = \cup_{i\in I} A_i, A_i \in \mathcal{X} \times \mathcal{Y}, I : \text{finite}\}.$$

Since $\{(x, y)\} = \{x\} \times \{y\} \in \mathcal{X} \times \mathcal{Y}$ for every $x \in X, y \in Y$, we have the next proposition.

Proposition 4. *If $\mathcal{X} = 2^X$ and $\mathcal{Y} = 2^Y$, then $\overline{\mathcal{X} \times \mathcal{Y}} = 2^{X\times Y}$.*

It follows from Proposition 4 that every function $f : X \times Y \to [0, 1]$ is $\overline{\mathcal{X} \times \mathcal{Y}}$-measurable. Let $A \in \overline{\mathcal{X} \times \mathcal{Y}}$. Then A can be represented as

$$A = \cup_{i\in I, j\in J}\{x_i\} \times \{y_j\} \tag{1}$$

for I, J: finite sets. Therefore we can define a fuzzy measure m on $\overline{\mathcal{X} \times \mathcal{Y}}$ by

$$m(A) := \perp_{i\in I, j\in J} \mu(\{x_i\}) \boxdot \nu(\{y_j\}). \tag{2}$$

The next prosition follows from the definition above immediately.

Proposition 5. *The extended fuzzy measure m on $\overline{\mathcal{X} \times \mathcal{Y}}$ is normal $\perp$-decomposable.*

Let $f : X \times Y \to [0, 1]$ be $\overline{\mathcal{X} \times \mathcal{Y}}$-measurable. f can be represented as

$$f := \perp_{i=1}^{n} r_{ij} 1_{\{x_i\}\times\{y_j\}} \text{ where } x_i \in X, y_j \in Y,$$

then it follows from t-conorm integral that

$$\begin{aligned}(T)\int f \boxdot dm &= \perp_{i,j} r_{ij} \boxdot m(\{(x_i, y_j)\}) = \perp_{i,j} r_{ij} \boxdot \mu(\{x_i\}) \boxdot \nu(\{y_j\}))\\ &= \perp_j (\perp_i r_{ij} \boxdot \mu(\{x_i\})) \boxdot \nu(\{y_j\}))\\ &= \perp_j ((T)\int f \boxdot d\mu) \boxdot \nu(\{y_j\}) = (T)\int (\int f \boxdot d\mu) \boxdot d\nu\end{aligned}$$

Therefore we have the next theorem.

Theorem 3. *Let $(X, \mathcal{X}, \mu)$ and $(Y, \mathcal{Y}, \nu)$ be fuzzy measure spaces and μ and ν be normal $\perp$-decomposable fuzzy measures. If $f : X \times Y \to [0,1]$ be a $\overline{\mathcal{X} \times \mathcal{Y}}$-mesurable function, then there exists a fuzzy measure m on $\overline{\mathcal{X} \times \mathcal{Y}}$ such that*

$$(T)\int((T)\int f d\mu)d\nu = (T)\int f dm = (T)\int((T)\int f d\nu)d\mu.$$

Considering $\perp = \hat{+}$, $\boxdot = \cdot$, the integral is standard Lebesgue integral. Theorem 3 is a part of standard Fubini's theorem.

Considering $\perp = \vee$, $\boxdot = \wedge$, and considering the characteristic function of $A \times B \in \mathcal{X} \times \mathcal{Y}$, we have $m(A \times B) = \vee_{a \in A, b \in B} \mu(\{a\}) \wedge \nu(\{b\})$. Therefore m is a possibility measure.

Corollary 3. *[10] Let $(X, \mathcal{X}, \mu)$ and $(Y, \mathcal{Y}, \nu)$ be two fuzzy measure spaces and μ and ν be possibility measures. Suppose that f be a $\overline{\mathcal{X} \times \mathcal{Y}}$-measurable function.*

Then, there exists a possibility measure m on $\overline{\mathcal{X} \times \mathcal{Y}}$ such that

$$(S)\int((S)\int f(x,y)d\mu)d\nu = \quad (S)\int((S)\int f(x,y)d\nu)d\mu = (S)\int f(x,y)dm$$

5 Citation Analysis and Multidimensional Integrals

As we have stated in the introduction, the results presented here were partially motivated by our interest on indexes for citation analysis. In a recent paper [12], we have shown that the new Hirsch index [4] is equivalent to a Sugeno integral with a particular fuzzy measure.

Formally, we consider for each researcher a function f that corresponds to the number of citations of a particular paper and a fuzzy measure μ on the sets of publications of that author. Then, the Sugeno integral of f with respect to μ corresponds to the Hirsch index when μ is the cardinality of the set. The paper also proves that the Choquet integral of function f with respect to the same measure corresponds to the standard method of counting the number of citations. See [12] for details.

From the perspective of [12], we can define new indexs for citation analysis using other fuzzy measures or other fuzzy integrals. In particular, we have considered in [10] the case that the function f is two dimensional: $f_a(x)$ corresponds to the number of citations of a paper x in year a. Also, we consider the function $g_a(x)$ that corresponds to the citations obtained by x in year a. Naturally, $f_\gamma(x) = \sum_{y \leq \gamma} g_y(x)$. Table 1 illustrates function g on five publications $X = \{p_1, p_2, p_3, p_4, p_5\}$.

In this context, it is relevant to consider the multidimensional integral of function g. Theorem 2 implies that the order in which we integrate g is not relevant (when appropriate conditions hold).

At this point, it is important to note that the measure μ involved in the Hirsch index is additive and with a range beyond $[0,1]$. So, the results obtained here are not directly applicable into that setting. Nevertheless, as the fuzzy integrals are monotonic with respect to the fuzzy measure, it is always possible to define

Table 1. Set of functions corresponding to the new number of citations for each year in the period 2002-2006 and for the following papers: $X = \{p_1, p_2, p_3, p_4, p_5\}$

	p_1	p_2	p_3	p_4	p_5	p_6
g_{2006}	60	30	5	4	2	1
g_{2005}	30	0	2	1	0	0
g_{2004}	8	20	0	0	0	0
g_{2003}	2	0	0	0	0	0
g_{2002}	0	0	0	0	0	0

$\mu' := \mu/K$ so that we have a normalized measure μ'. Any comparison between the *performance* of two researchers will be not influenced by a normalization factor K (if both use the same value K).

In particular, when the Choquet integral is considered, the integral with respect to $\mu(A) = |A|/K_1$ and $\nu(A) = |A|/K_2$ corresponds to:

$$\frac{\sum_i \sum_{x \in X} g_i(x)}{K_1 K_2}$$

So, the integral of g corresponds to the total number of citations (multiplied by a normalization factor). In this case, according to the results above, we have $m(C) = (|A|/K_1)(|B|/K_2)$, when $C = A \times B$. When we consider the results on the extension of the domain, we have that with $\mu(\{r\}) = 1/K_1$ and $\nu(\{s\}) = 1/K_2$ the same result is obtained. This corresponds to the Choquet integral of the same function with $m(C) = |C|/(K_1 K_2)$. Note that this measure is consistent with $\mu(C) = (|A|/K_1)(|B|/K_2)$, when $C = A \times B$.

Note that for the particular function g in Table 1, only Theorem 3 applies but not Theorem 2. This is so because the function g is not $\mathcal{X} \times \mathcal{Y}$-measurable but it is $\overline{\mathcal{X} \times \mathcal{Y}}$-measurable. For example, $\{(p_1, 2006), (p_2, 2006), (p_3, 2005)\}$ is not in $\mathcal{X} \times \mathcal{Y}$ but is in $\overline{\mathcal{X} \times \mathcal{Y}} = 2^{X \times Y}$.

In the case of the Sugeno integral, Corollary 2 also shows that the result of the Sugeno integral for $\mathcal{X} \times \mathcal{Y}$-measurable functions does not depend on the order in which we integrate and that, with $\mu(A) = |A|/K_1$ and $\nu(A) = |A|/K_2$ the multidimensional integral is equivalent to one with respect to m defined by $m(C) = \min(|A|/K_1, |B|/K_2)$, for those $C = A \times B$. Note that Theorem 2 does not apply to the function g above because g is not $\mathcal{X} \times \mathcal{Y}$-measurable. Besides, as m is not possibility measures, it is not possible to apply Theorem 3 on the extension.

6 Conclusions

In this paper we have studied multidimensional fuzzy integrals and briefly described their application to citation analsysis.

Acknowledgements

Partial support by Generalitat de Catalunya (AGAUR, 2006BE-2 00338) and by the Spanish MEC (project "PROPRIETAS", SEG2004-04352-C04-02) is acknowledged.

References

1. Benvenuti, P., Mesiar, R.: A note on Sugeno and Choquet integrals. In: Proc. 8th Intern. Conf. Information Processing and Management of Uncertainty in Knowledge-based Systems (IPMU 2000), pp. 582–585 (2000)
2. Choquet, G.: Theory of capacities. Ann. Inst. Fourier 5, 131–295 (1953/54)
3. Dellacherie, C.: Quelques commentaires sur les prolongements de capacités, Séminaire de Probabilités, 1969/1970, Strasbourg, Lecture Notes in Mathematics, vol. 191, pp. 77–81(1971)
4. Hirsch, J.E.: An index to quantify an individual's scientific research output. In: Proc. of the National Academy of Sciences, vol. 102(45), pp. 16569–16572 (2005)
5. Klement, E.P., Mesiar, R., Pap, E.: Triangular Norms. Kluwer Academic Publishers, Dordrecht (2000)
6. Klement, E.P., Mesiar, R., Pap, E.: Integration with respect to decomposable measures, based on a conditionally distributive semiring on the unit interval. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 8(6), 701–717 (2000)
7. Ling, C.H.: Representation of associative functions. Publ. Math. Debrecen 12, 189–212 (1965)
8. Machida, M.: A study on non-additive set functions, Master thesis, Tokyo Institute of Technology 1990 (in Japanese)
9. Murofushi, T., Sugeno, M.: An interpretation of fuzzy measures and the Choquet integral as an integral with respect to a fuzzy measure. Fuzzy Sets and Systems 29, 201–227 (1989)
10. Narukawa, Y., Torra, V.: Multidimensional integrals and citation analysis. In: Proc. of EUROFUSE 2007, pp. 185–189 (2007)
11. Sugeno, M.: Theory of fuzzy integrals and its applications, Ph. D. Dissertation, Tokyo Institute of Technology, Tokyo, Japan (1974)
12. Torra, V., Narukawa, Y.: On citation indices: Choquet and Sugeno based indices. IEEE transaction on Fuzzy systems (to appear)
13. Yager, R.R., Rybalov, A.: Uninorm aggregation operators. Fuzzy sets and Systems 80, 111–120 (1996)

Lindig's Algorithm for Concept Lattices over Graded Attributes*

Radim Belohlavek[1,3], Bernard De Baets[2], Jan Outrata[3], and Vilem Vychodil[3]

[1] Dept. Systems Science and Industrial Engineering
T. J. Watson School of Engineering and Applied Science
Binghamton University–SUNY, PO Box 6000, Binghamton, NY 13902–6000, USA
rbelohla@binghamton.edu
[2] Dept. Appl. Math., Biometrics, and Process Control, Ghent University
Coupure links 653, B-9000 Gent, Belgium
bernard.debaets@ugent.be
[3] Dept. Computer Science, Palacky University, Olomouc
Tomkova 40, CZ-779 00 Olomouc, Czech Republic
{jan.outrata, vilem.vychodil}@upol.cz

Abstract. Formal concept analysis (FCA) is a method of exploratory data analysis. The data is in the form of a table describing relationship between objects (rows) and attributes (columns), where table entries are grades representing degrees to which objects have attributes. The main output of FCA is a hierarchical structure (so-called concept lattice) of conceptual clusters (so-called formal concepts) present in the data. This paper focuses on algorithmic aspects of FCA of data with graded attributes. Namely, we focus on the problem of generating efficiently all clusters present in the data together with their subconcept-superconcept hierarchy. We present theoretical foundations, the algorithm, analysis of its efficiency, and comparison with other algorithms.

1 Introduction

Our paper contributes to the area of exploratory analysis of tabular data. Namely, we focus on data supplied as tables with rows corresponding to objects and columns corresponding to attributes. Datasets of this form are often described by bivalent (presence/absence) attributes. That is, each attribute either applies or does not apply to a particular object. The corresponding data table is thus binary, i.e., it is a matrix filled with 0's (the object given by row does not have the attribute given by column) and 1's (the object given by row has the attribute given by column). Formal concept analysis (FCA) [8,20] aims at revealing all "conceptual clusters" (so-called formal concepts) which are hidden in the binary data table. Formal concepts are particular clusters of objects and

* Supported by Kontakt 1–2006–33 (Bilateral Scientific Cooperation, project "Algebraic, logical and computational aspects of fuzzy relational modelling paradigms"), by grant No. 1ET101370417 of GA AV ČR, by grant No. 201/05/0079 of the Czech Science Foundation, and by institutional support, research plan MSM 6198959214.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 156–167, 2007.

attributes that correspond to maximal submatrices filled with 1's. Alternatively, formal concepts may be understood as concepts in the traditional sense—as entities consisting of a set A of objects and a set B of attributes to which the concept applies (e.g., concept dog applies to objects poodle, foxhound, ... and attributes barks, has limbs, ...). Formal concepts are partially ordered by the subconcept-superconcept hierarchy. The resulting partially ordered set of formal concepts (the so-called concept lattice) represents a hierarchical structure of all naturally interpretable clusters (concepts) existing in the data (e.g., concept mammal is more general a concept than dog). Applications of FCA in exploratory data analysis can be found in [6,8]; [8] provides theoretical foundations.

In practice, more often than not, attributes are graded (fuzzy) rather than bivalent. That is, an attribute applies to an object to a certain degree which may be represented, e.g., by a number from the unit interval $[0, 1]$. The data table is then a $[0, 1]$-valued matrix, with table entries corresponding to degrees to which attributes apply to objects. There have been several approaches to FCA in a graded setting. The most relevant approach was independently developed in [1,2,4] and [19]. Up to now, not much attention has been paid to computational aspects of FCA with graded attributes. As an exception, in [3] the author presents an algorithm for determining formal concepts present in data table with graded attributes.

The aim of this paper is to propose another algorithm for generating all formal concepts (which can be seen again as certain maximal submatrices). Unlike the algorithm presented in [3], our algorithm enables us to generate all formal concepts together with their subconcept-superconcept hierarchy. The absence of conceptual hierarchy is not crucial if, for instance, the output of FCA is used for preprocessing (e.g., for mining non-redundant association rules, see [21]). On the other hand, if we want to present the output of analysis directly to users, it is more convenient to depict the clusters in a hierarchy which models the natural subconcept-superconcept ordering. Thus, from the point of view of applications, it is important to have an efficient algorithm which generates the hierarchy along with the clusters present in data table with graded attributes.

The paper is organized as follows. In Section 2, we present preliminaries from classical formal concept analysis, fuzzy sets, and formal concept analysis of data with graded attributes. In Section 3, we present the algorithm and prove its correctness. Section 4 contains experiments and comparisons of the new algorithm with that from [3].

2 Preliminaries

This section provides basic notions of FCA and fuzzy logic. More details can be found in [6,8,20] (formal concept analysis) and in [1,9,12,16] (fuzzy logic).

2.1 Formal Concept Analysis

Let X and Y be nonempty sets (of objects and attributes, respectively), let $I \subseteq X \times Y$ be a binary relation between X and Y. The triplet $\langle X, Y, I \rangle$ is called a *formal context,* the fact $\langle x, y \rangle \in I$ is interpreted as "x has y" ("y applies

to x"). A formal context corresponds to a binary data table with rows and columns corresponding to objects and attributes, respectively, such that the entry corresponding to objects x and attribute y is 1 if $\langle x, y\rangle \in I$ and 0 if $\langle x, y\rangle \notin I$. For $A \subseteq X$ and $B \subseteq Y$ we define sets $A^{\uparrow} \subseteq Y$ and $B^{\downarrow} \subseteq X$ by

$$A^{\uparrow} = \{y \in Y \mid \text{for each } x \in A\colon \langle x, y\rangle \in I\}, \tag{1}$$

$$B^{\downarrow} = \{x \in X \mid \text{for each } y \in B\colon \langle x, y\rangle \in I\}. \tag{2}$$

Therefore, $A^{\uparrow}$ is the set of all attributes common to all objects from A and $B^{\downarrow}$ is the set of all objects common to all attributes from B. According to the traditional understanding, a pair $\langle A, B\rangle$ where $A \subseteq X$ (so-called *extent*) and $B \subseteq Y$ (so-called *intent*) is called a *formal concept of* $\langle X, Y, I\rangle$ iff $A^{\uparrow} = B$ and $B^{\downarrow} = A$. Thus, $\langle A, B\rangle$ is a formal concept of $\langle X, Y, I\rangle$ iff A is the set of all objects sharing all the attributes of B and, conversely, B is the set of all attributes common to all objects from A. Alternatively, formal concepts may be understood as maximal rectangles of the data matrix which are filled with 1's: $\langle A, B\rangle$ is a formal concept of $\langle X, Y, I\rangle$ iff it is a maximal rectangle filled with 1's which is contained in I (i.e., a maximal submatrix of $\langle X, Y, I\rangle$ filled with 1's).

The set $\mathcal{B}(X, Y, I) = \{\langle A, B\rangle \mid A^{\uparrow} = B,\ B^{\downarrow} = A\}$ is called the concept lattice induced by the input data $\langle X, Y, I\rangle$. Moreover, $\mathcal{B}(X, Y, I)$ can be equipped by a partial order relation $\leq$ defined by $\langle A, B\rangle \leq \langle C, D\rangle$ iff $A \subseteq C$ (or, equivalently, $B \supseteq D$). The partial order $\leq$ which is, in fact, a complete lattice order, models the subconcept-superconcept hierarchy: $\langle A, B\rangle \leq \langle C, D\rangle$ means that the concept $\langle C, D\rangle$ is more general than $\langle A, B\rangle$ (covers more objects, or, equivalently, less attributes).

2.2 Fuzzy Sets and Fuzzy Relations

A fuzzy set A in a universe set X [22] is a mapping assigning to each $x \in X$ a truth degree $A(x) \in L$ where L is some partially ordered set of truth degrees containing at least 0 (full falsity) and 1 (full truth). Usually, L is the unit interval $[0, 1]$ or a suitable subset of $[0, 1]$. $A(x)$ is interpreted as the degree to which x belongs to A. The notion of a fuzzy set enables us to model vaguely (nonsharply) delineated collections: For instance, the collection described linguistically as "tall men" can be modeled by a fuzzy set to which men with heights 150, 180, and 200 cm belong to degrees 0, 0.7, and 1, respectively.

In order to be able to develop the basic calculus with fuzzy sets and fuzzy relations, the set L of truth degrees needs to be equipped by suitable operations generalizing logical connectives of classical (two-valued) logic. Particularly, we will need fuzzy conjunction $\otimes$ and fuzzy implication $\rightarrow$. In the literature, there have been proposed several fuzzy conjunctions and fuzzy implications [16]. A general class of logical connectives is captured by the notion of a complete residuated lattice [1,10,14]: A complete residuated lattice is a structure $\mathbf{L} = \langle L, \wedge, \vee, \otimes, \rightarrow, 0, 1\rangle$ such that (1) $\langle L, \wedge, \vee, 0, 1\rangle$ is a complete lattice (with the least element 0, greatest element 1), i.e. a partially ordered set in which arbitrary infima ($\bigwedge$) and suprema ($\bigvee$) exist; (2) $\langle L, \otimes, 1\rangle$ is a commutative monoid, i.e. $\otimes$ is a binary operation which is commutative, associative, and $x \otimes 1 = x$ ($x \in L$); (3) $\otimes, \rightarrow$

satisfy $x \otimes y \leq z$ iff $x \leq y \to z$. Operations $\otimes$ (multiplication) and $\to$ (residuum) play the role of a fuzzy conjunction and a fuzzy implication, respectively. The class of complete residuated lattices includes structures defined on the real unit interval with $\wedge$ and $\vee$ being minimum and maximum, respectively, $\otimes$ being left-continuous t-norm, and $\to$ being its residuum, see [1,9,12,15] for details.

In what follows, $\mathbf{L}$ always refers to a complete residuated lattice and $\leq$ denotes the induced lattice order (i.e., $a \leq b$ iff $a \wedge b = a$ iff $a \vee b = b$ iff $a \to b = 1$). We write $a < b$ to denote that $a \leq b$ and $a \neq b$. Given $\mathbf{L}$, a fuzzy set with truth degrees from $\mathbf{L}$ (called also an $\mathbf{L}$-set) is a mapping $A\colon X \to L$ assigning to any $x \in X$ a truth degree $A(x) \in L$ to which x belongs to A. Similarly, a binary fuzzy relation R with truth degrees from $\mathbf{L}$ is a mapping $R\colon X \times Y \to L$ assigning to any $x \in X$ and $y \in Y$ a truth degree $R(x, y) \in L$ to which x and y are related under R. The set of all $\mathbf{L}$-sets in a universe X is denoted L^X. For a fuzzy set $A \in L^X$ and a truth degree $a \in L$ we denote by ${}^a\!/\!A$ the a-cut of A, i.e. ${}^a\!/\!A = \{x \in X \,|\, A(x) \geq a\}$ (the ordinary set of elements from X which belong to A to degree at least a). A fuzzy set $A \in L^X$ is called crisp if, for each $x \in X$, $A(x) \in \{0, 1\}$. Following common usage we will identify crisp fuzzy sets in X with (characteristic functions of) ordinary subsets of X. In particular, by $\emptyset$ and X we denote crisp fuzzy sets $\emptyset \in L^X$ and $X \in L^X$ such that $\emptyset(x) = 0$ and $X(x) = 1$ for each $x \in X$. For fuzzy sets $A, B \in L^X$ we put $A \subseteq B$ (A is a subset of B) if for each $x \in X$ we have $A(x) \leq B(x)$, in which case we say that A is (fully) contained in B. If for $A, B \in L^X$ we have $A \subseteq B$ and there is $x \in X$ such that $A(x) < B(x)$, we write $A \subset B$ and say that A is strictly contained in B.

2.3 Fuzzy Attributes, Fuzzy Contexts, and Formal Concepts

When dealing with real-world situations, it is very often the case that attributes that we observe on the objects of interest are fuzzy rather than bivalent. In general, an attribute y applies to an object x to some degree $I(x, y) \in L$ not necessarily being equal to 0 or 1. The larger $I(x, y)$, the more y applies to x. From the point of view of FCA, the input data table, which is no longer a binary one, is represented by a triplet $\langle X, Y, I\rangle$ (called a *formal fuzzy context*) where $I \in L^{X\times Y}$, i.e. I is a fuzzy relation between X and Y.

The agenda of formal concept analysis of data with fuzzy attributes [1,4,19] is the following. For fuzzy sets $A \in L^X$ (i.e., A is a fuzzy set of objects) and $B \in L^Y$ (i.e., B is a fuzzy set of attributes), consider fuzzy sets $A^\uparrow \in L^Y$ (fuzzy set of attributes) and $B^\downarrow \in L^X$ (fuzzy set of objects) defined by

$$A^\uparrow(y) = \bigwedge\nolimits_{x\in X}(A(x) \to I(x, y)), \tag{3}$$

$$B^\downarrow(x) = \bigwedge\nolimits_{y\in Y}(B(y) \to I(x, y)). \tag{4}$$

Using basic rules of fuzzy logic, one can see that $A^\uparrow(y)$ is the truth degree of "y is shared by all objects from A" and $B^\downarrow(x)$ is the truth degree of "x has all attributes from B", i.e. (3) and (4) properly generalize (1) and (2). Each $\langle A, B\rangle \in L^X \times L^Y$ such that $A^\uparrow = B$ and $B^\downarrow = A$ is called a *formal fuzzy concept of* $\langle X, Y, I\rangle$. The set of all formal fuzzy concepts of $\langle X, Y, I\rangle$ will be denoted by $\mathcal{B}(X, Y, I)$. Both the *extent* A and *intent* B of a formal fuzzy concept

$\langle A, B\rangle \in \mathcal{B}(X, Y, I)$ are fuzzy sets. This corresponds well to the intuition that a concept may apply to objects and attributes to various intermediate degrees, not necessarily to 0 and 1 only. For brevity, by $\mathrm{Int}(X, Y, I)$ we denote the set of all intents of $\langle X, Y, I\rangle$, i.e. $\mathrm{Int}(X, Y, I) = \{B \in L^Y \mid \langle A, B\rangle \in \mathcal{B}(X, Y, I) \text{ for some } A \in L^X\}$. Analogously, $\mathrm{Ext}(X, Y, I)$ denotes the set of all extents of $\langle X, Y, I\rangle$. The conceptual hierarchy in $\mathcal{B}(X, Y, I)$ is modeled by a relation $\leq$ defined on $\mathcal{B}(X, Y, I)$ by

$$\langle A_1, B_1\rangle \leq \langle A_2, B_2\rangle \quad \text{iff} \quad A_1 \subseteq A_2 \ (\text{iff } B_1 \supseteq B_2), \tag{5}$$

where $A_1 \subseteq A_2$ means that A_1 is fully contained in A_2 for each $x \in X$ (see Section 2.2). The following theorem characterizes the structure of fuzzy concept lattices.

Theorem 1 (see [4]). *The set $\mathcal{B}(X, Y, I)$ is under $\leq$ a complete lattice where the infima and suprema are given by*

$$\textstyle\bigwedge_{j\in J}\langle A_j, B_j\rangle = \langle \bigcap_{j\in J} A_j, (\bigcup_{j\in J} B_j)^{\downarrow\uparrow}\rangle, \tag{6}$$

$$\textstyle\bigvee_{j\in J}\langle A_j, B_j\rangle = \langle (\bigcup_{j\in J} A_j)^{\uparrow\downarrow}, \bigcap_{j\in J} B_j\rangle. \tag{7}$$

Moreover, an arbitrary complete lattice $\mathbf{V} = \langle V, \wedge, \vee\rangle$ is isomorphic to some $\mathcal{B}(X, Y, I)$ iff there are mappings $\gamma\colon X \times L \to V$, $\mu\colon Y \times L \to V$ such that $\gamma(X, L)$ is $\bigwedge$-dense in V; $\mu(Y, L)$ is $\bigvee$-dense in V; $a \otimes b \leq I(x, y)$ iff $\gamma(x, a) \leq \mu(y, b)$. □

If we take $\mathbf{L}$ with $L = \{0, 1\}$, i.e. there are only two truth degrees involved (our structure of truth degrees is the two-valued Boolean algebra), all notions introduced in Section 2.1 will become particular cases of notions presented in this section. This is the way the fuzzy approach generalizes the classical one [8].

Remark 1. Formal fuzzy concepts can also be characterized as maximal rectangles contained in I: For a pair $\langle A, B\rangle \in L^X \times L^Y$ (call it a rectangle), define a fuzzy relation $A \otimes B \in L^{X\times Y}$ by $(A \otimes B)(x, y) = A(x) \otimes B(y)$. $\langle A, B\rangle$ is said to be contained in I if $A \otimes B \subseteq I$. Furthermore, put $\langle A, B\rangle \sqsubseteq \langle A', B'\rangle$ iff $A \subseteq A'$ and $B \subseteq B'$. Then we have [1] that $\langle A, B\rangle$ is a formal fuzzy concept of $\langle X, Y, I\rangle$ iff it is a maximal (w.r.t. $\sqsubseteq$) rectangle contained in I.

3 Computing Fuzzy Concepts and Conceptual Hierarchy

The best known algorithm for computing formal concepts is probably Ganter's NextIntent algorithm, see [7,8]. The original NextIntent generates in a lexical order all concepts present in a (classical/bivalent) context. Graded extension of Ganter's algorithm has been presented in [3]. In this section we develop an algorithm for computing fuzzy concepts (along with their hierarchy) which is inspired by the Lindig's NextNeighbor algorithm [18].

Our goal is the following. Given an input data table represented by a formal fuzzy context $\langle X, Y, I\rangle$, (i) generate all formal fuzzy concepts $\langle A, B\rangle$ of the fuzzy concept lattice $\mathcal{B}(X, Y, I)$ and, at the same time; (ii) compute for each fuzzy concept $\langle A, B\rangle$ a set of its direct subconcepts and direct superconcepts. The sets of (direct) subconcepts/superconcepts fully determine the whole hierarchical structure of fuzzy concepts. Information about direct subconcepts and superconcepts

is of crucial importance for applications. For instance, it allows us to navigate users through the concepts, it is used as an input for geometrical method for drawing concept lattices, and more (see [8] for other applications).

Each fuzzy concept $\langle A, B\rangle$ is uniquely given by each of its components: by its extent A (since $B = A^{\uparrow}$) or by its intent B (since $A = B^{\downarrow}$). Therefore, in order to generate all fuzzy concepts from $\mathcal{B}(X, Y, I)$, it is sufficient to generate all intents from $\mathrm{Int}(X, Y, I)$ (or, equivalently, all extents). Moreover, $B \in L^Y$ is an intent from $\mathrm{Int}(X, Y, I)$ iff $B = B^{\downarrow\uparrow}$, i.e. iff B is a fixed point of the fuzzy closure operator ${}^{\downarrow\uparrow}\colon L^Y \to L^Y$ which is a composition of operators ${}^{\downarrow}$ and ${}^{\uparrow}$ defined by (3) and (4), see [1]. Thus, the task to compute all fuzzy concepts present in $\langle X, Y, I\rangle$ can be reduced to a task of computing all fixed points of a fuzzy closure operator ${}^{\downarrow\uparrow}\colon L^Y \to L^Y$. In what follows we forget about the operators ${}^{\downarrow}, {}^{\uparrow}$ for a while and develop the algorithm so that it accepts a general fuzzy closure operator $\mathrm{C}\colon L^Y \to L^Y$ (i.e., C is extensive, monotone, and idempotent [1]) as its input, and produces a set of its fixed points (with their hierarchy) as its output. The set of all fixed points of $\mathrm{C}\colon L^Y \to L^Y$ will be denoted by $\mathrm{fix}(\mathrm{C})$, i.e.

$$\mathrm{fix}(\mathrm{C}) = \{B \in L^Y \mid B = \mathrm{C}(B)\} = \{\mathrm{C}(B) \mid B \in L^Y\}. \tag{8}$$

Note that due to computational reasons, we restrict ourselves to finite structures of truth degrees, and always assume that X (set of objects) and Y (set of attributes) are finite. This will ensure that the set of all fuzzy concepts extracted from the data will be finite and thus enumerable in finitely many steps. For simplicity, we describe only the case when $\mathbf{L}$ is linearly ordered (the case of general finite $\mathbf{L}$ is technically more complicated and will be discussed in a full version of this paper). In the rest of the paper $\mathrm{C}\colon L^Y \to L^Y$ always denotes a fuzzy closure operator.

For convenience, denote $L = \{a_1, \ldots, a_k\}$ so that $a_1 < a_2 < \cdots < a_k$. If $i < k$, we write a_i^+ instead of a_{i+1}. Upper neighbors can be introduced as follows: $D \in \mathrm{fix}(\mathrm{C})$ is called an *upper neighbor of* $B \in \mathrm{fix}(\mathrm{C})$ (*w.r.t.* C), written $B \prec^{\mathrm{C}} D$, if (i) $B \subset D$, and (ii) there is no $D' \in \mathrm{fix}(\mathrm{C})$ such that $B \subset D' \subset D$.

Lower neighbors can be defined dually. Note that if C is ${}^{\downarrow\uparrow}$, then upper neighbors of an intent B with respect to ${}^{\downarrow\uparrow}$ are exactly the intents of the direct subconcepts of $\langle B^{\downarrow}, B\rangle$. One should not be mislead here: even if the upper neighbors of B are intents which are greater than B, they determine *subconcepts* due to the fact that greater fuzzy sets of attributes are shared by smaller fuzzy sets of objects, see the definition of conceptual hierarchy (5).

For each $B \in L^Y$ and $y \in Y$ such that $B(y) < 1$, let $\mathrm{C}\big(B \cup \{{}^{B(y)^+}\!/_y\}\big)$ be abbreviated by $[y]^{\mathrm{C}}_B$. From now on, if we write $[y]^{\mathrm{C}}_B$ we tacitly assume that $B(y) < 1$. If $[y]^{\mathrm{C}}_B$ is an upper neighbor of B w.r.t. C, then $[y]^{\mathrm{C}}_B$ will be called an upper neighbor generated by y; y is called a generator of $[y]^{\mathrm{C}}_B$. For technical reasons, we assume $Y = \{y_1, \ldots, y_n\}$ and consider a fixed order of attributes from Y given by the indices, i.e. $y_i < y_j$ iff $i < j$. In the sequel, we write just $i < j$ to denote $y_i < y_j$. The following assertion says that each upper neighbor is in fact an upper neighbor generated by some y. In addition the generator y can be chosen so that it is the greatest generator with respect to the ordering of attributes.

Lemma 1. *The following are true for any fuzzy closure operator* $\mathrm{C}\colon L^Y \to L^Y$.

(i) *For each* $B \in L^Y$ *and* $y \in Y$ *such that* $B(y) < 1$, *we have* $B(y) < ([y]_B^{\mathrm{C}})(y)$.

(ii) *Let* $B, D \in \mathrm{fix}(\mathrm{C})$ *such that* $B \subset D$ *and put*

$$i = \max\{j \mid B(y_j) < D(y_j)\}. \tag{9}$$

Then, for each $k > i$, $D(y_k) = B(y_k)$. *Moreover, if* $B \prec^{\mathrm{C}} D$ *then* $D = [y_i]_B^{\mathrm{C}}$.

Proof. (i) is obvious. In order to prove (ii), let $B, D \in \mathrm{fix}(\mathrm{C})$ such that $B \subset D$. Clearly, $\{j \mid B(y_j) < D(y_j)\}$ is a nonempty finite set, i.e. it has a maximum. Denote the maximum by i as in (9). Take any $k > i$. Since $B \subset D$ and i is the maximum of all indices such that $B(y_j) < D(y_j)$, we get $B(y_k) = D(y_k)$. Suppose we have $B \prec^{\mathrm{C}} D$. From $B(y_i) < B(y_i)^+ \leq D(y_i)$, we get $B \cup \{{}^{B(y_i)^+}\!/_{y_i}\} \subseteq D$, which further gives $[y_i]_B^{\mathrm{C}} = \mathrm{C}\big(B \cup \{{}^{B(y_i)^+}\!/_{y_i}\}\big) \subseteq \mathrm{C}(D) = D$ because D is a fixed point of C. Furthermore, from $B \subset [y_i]_B^{\mathrm{C}} \subseteq D$ it follows that $[y_i]_B^{\mathrm{C}} = D$ because $[y_i]_B^{\mathrm{C}} \in \mathrm{fix}(\mathrm{C})$ and D was supposed to be an upper neighbor of B. □

Indices defined by (9) will play an important role. Therefore, we will introduce the following notation. For $\mathrm{C}\colon L^Y \to L^Y$, $B, D \in \mathrm{fix}(\mathrm{C})$ such that $B \subset D$, and i given by (9), we denote $y_i \in Y$ by $y_B^{\mathrm{C}}(D)$. Furthermore, we put

$$\mathcal{M}_B^{\mathrm{C}} = \{y \in Y \mid B \prec^{\mathrm{C}} [y]_B^{\mathrm{C}} \text{ and } y = y_B^{\mathrm{C}}([y]_B^{\mathrm{C}})\}. \tag{10}$$

Note that y_B^{C} and, consequently, $\mathcal{M}_B^{\mathrm{C}}$ depend on the chosen ordering of attributes. Since the ordering is fixed, we will not mention it explicitly. Regardless of the ordering, we have that $\{[y]_B^{\mathrm{C}} \mid y \in \mathcal{M}_B^{\mathrm{C}}\}$ is a set of all upper neighbors of B w.r.t. C. Moreover, the attributes from $\mathcal{M}_B^{\mathrm{C}}$ uniquely correspond to the upper neighbors from $\{[y]_B^{\mathrm{C}} \mid y \in \mathcal{M}_B^{\mathrm{C}}\}$. This follows directly from (10) and from Lemma 1(ii). Hence, in order to compute the upper neighbors it suffices to determine $\mathcal{M}_B^{\mathrm{C}}$. The following assertions provides us with a quick test of presence of an attribute in $\mathcal{M}_B^{\mathrm{C}}$.

Theorem 2. *Let* $B \in \mathrm{fix}(\mathrm{C})$ *and* $y_i \in Y$ *such that* $y_i = y_B^{\mathrm{C}}([y_i]_B^{\mathrm{C}})$. *Then we have* $y_i \in \mathcal{M}_B^{\mathrm{C}}$ *iff for each* $y_k \in \mathcal{M}_B^{\mathrm{C}}$ *such that* $k < i$ *we have* $([y_i]_B^{\mathrm{C}})(y_k) = B(y_k)$.

Proof. Take $B \in \mathrm{fix}(\mathrm{C})$ and let $y_i \in Y$ such that $y_i = y_B^{\mathrm{C}}([y_i]_B^{\mathrm{C}})$. Hence, from (9) it follows that $([y_i]_B^{\mathrm{C}})(y_i) > B(y_i)$ and $([y_i]_B^{\mathrm{C}})(y_k) = B(y_k)$ $(k > i)$.

"⇒": Let $y_i \in \mathcal{M}_B^{\mathrm{C}}$, i.e. $B \prec^{\mathrm{C}} [y_i]_B^{\mathrm{C}}$. Take any $y_k \in \mathcal{M}_B^{\mathrm{C}}$ such that $k < i$. Suppose, by contradiction, that $([y_i]_B^{\mathrm{C}})(y_k) > B(y_k)$. Since we have

$$B(y_k) < B(y_k)^+ \leq ([y_i]_B^{\mathrm{C}})(y_k),$$

we get $B \cup \{{}^{B(y_k)^+}\!/_{y_k}\} \subseteq [y_i]_B^{\mathrm{C}}$, i.e.

$$B \subset [y_k]_B^{\mathrm{C}} = \mathrm{C}\big(B \cup \{{}^{B(y_k)^+}\!/_{y_k}\}\big) \subseteq \mathrm{C}\big([y_i]_B^{\mathrm{C}}\big) = [y_i]_B^{\mathrm{C}}. \tag{11}$$

Since $B \prec^{\mathrm{C}} [y_i]_B^{\mathrm{C}}$, (11) yields $[y_k]_B^{\mathrm{C}} = [y_i]_B^{\mathrm{C}}$. From $y_k = y_B^{\mathrm{C}}([y_k]_B^{\mathrm{C}})$ and $k < i$ is follows that $([y_k]_B^{\mathrm{C}})(y_i) = B(y_i)$, which is a contradiction to $([y_k]_B^{\mathrm{C}})(y_i) = ([y_i]_B^{\mathrm{C}})(y_i) > B(y_i)$. Therefore, $([y_i]_B^{\mathrm{C}})(y_k) = B(y_k)$.

"⇐": Conversely, let $([y_i]_B^{\mathrm{C}})(y_k) = B(y_k)$ be true for each $y_k \in \mathcal{M}_B^{\mathrm{C}}$ such that $k < i$. We prove that $y_i \in \mathcal{M}_B^{\mathrm{C}}$. In order to prove this claim, it suffices to check

Algorithm 1. (Compute all upper neighbors of B w. r. t. C)

```
1  procedure NEIGHBORS (B, C):
2    U := ∅
3    Min := {y ∈ Y | B(y) < 1}
4    for each y ∈ Y such that B(y) < 1:
5      D := [y]_B^C
6      Increased := {z ∈ Y | z ≠ y and B(z) < D(z)}
7      if Min ∩ Increased = ∅:
8        add D to U
9      else:
10       remove y from Min
11   return U
```

that no $[y]_B^{\mathrm{C}}$ where $y_i \neq y \in \mathcal{M}_B^{\mathrm{C}}$ is contained in $[y_i]_B^{\mathrm{C}}$ because this will give $B \prec^{\mathrm{C}} [y_i]_B^{\mathrm{C}}$ from which the claim follows immediately. Thus, take $y_i \neq y \in \mathcal{M}_B^{\mathrm{C}}$. If $y = y_k$ where $k < i$ then, by assumption, $([y_i]_B^{\mathrm{C}})(y_k) = B(y_k)$ which directly gives that $[y_k]_B^{\mathrm{C}}$ cannot be contained in $[y_i]_B^{\mathrm{C}}$ because $([y_k]_B^{\mathrm{C}})(y_k) > B(y_k) = ([y_i]_B^{\mathrm{C}})(y_k)$. If $i < k$, we have $([y_i]_B^{\mathrm{C}})(y_k) = B(y_k)$ on account of $y_B^{\mathrm{C}}([y_i]_B^{\mathrm{C}}) = y_i$. Hence, again, $[y_k]_B^{\mathrm{C}}$ cannot be contained in $[y_i]_B^{\mathrm{C}}$ which proves $y_i \in \mathcal{M}_B^{\mathrm{C}}$. □

Theorem 2 leads to Algorithm 1 for computing all upper neighbors. The algorithm accepts $\mathrm{C}\colon L^Y \to L^Y$ and $B \in \mathrm{fix}(\mathrm{C})$ as its input and produces a set of all upper neighbors of B w.r.t C.

Theorem 3. *Algorithm 1 is correct.*

Proof. The algorithm uses the following variables: $\mathcal{U}$ is a set of upper neighbors which is initially empty; *Min* is an ordinary set of attributes which should be understood as a set of possible generators of upper neighbors. The roles of $\mathcal{U}$ and *Min* are the following. The set *Min* is initially set to $\{y \in Y \mid B(y) < 1\}$ and at the end of computation, we will have $Min = \mathcal{M}_B^{\mathrm{C}}$, i.e. *Min* will be a collection of generators of upper neighbors which is built by removing attributes which do not belong to (10), the ordering of attributes (see comments before Lemma 1) is given by the order in which the loop between lines 4–10 processes the attributes. As we can see from lines 5, 8 and 10, y is left in *Min* iff $D = [y]_B^{\mathrm{C}}$ is added to $\mathcal{U}$. The important part of the algorithm is the test present at line 7. It can be seen that $Min \cap Increased = \emptyset$ happens iff $y_B^{\mathrm{C}}(D) = y$ and for each $y_k \in Min$ that has already been processed (i.e., $k < i$), we have $D(y_k) = B(y_k)$. Thus, Theorem 2 gives that test at line 7 is successful iff D is an upper neighbor of B such that $y \in \mathcal{M}_B^{\mathrm{C}}$. Hence, y can be left in *Min* and D is added to $\mathcal{U}$ which is exactly what happens between lines 7–10. By induction, we can prove that at the end of computation, $Min = \mathcal{M}_B^{\mathrm{C}}$ and $\mathcal{U} = \{[y]_B^{\mathrm{C}} \mid y \in \mathcal{M}_B^{\mathrm{C}}\}$. The full proof is omitted due to the limited scope of this paper. □

Example 1. For illustration, take $\mathbf{L} = \langle\{0, 0.5, 1\}, \min, \max, \otimes, \to, 0, 1\rangle$ with $\otimes$ and $\to$ being Łukasiewicz operations. Consider a fuzzy context from Fig. 1 (left) and an induced closure operator C being $^{\downarrow\uparrow}$. Let $B = \{{}^{0.5}/c, {}^{0.5}/d, e\}$. When NEIGHBORS (B, C) is invoked, the procedure goes as follows. First, *Min* is set to

Algorithm 2. (Compute all fixed points of C and their hierarchy)

```
1  procedure GenerateFrom(B):
2    while B ≠ Y:
3      B* := Neighbors(B, C)
4      𝒩 := B* − ℱ
5      for each D ∈ B*:
6        add B to D*
7        if D ∈ 𝒩:
8          add D to ℱ
9      for each D ∈ 𝒩:
10       call GenerateFrom(D)
11 procedure Lattice(C, Y):
12   ℱ := ∅
13   B := C(∅)
14   add B to ℱ
15   call GenerateFrom(B)
16   return ⟨ℱ, {B* | B ∈ ℱ}, {B* | B ∈ ℱ}⟩
```

$\{a, b, c, d\}$. Then, $a \in Y$ is processed. We get $D = [a]_B^{\mathrm{C}} = \{^{0.5}/a, ^{0.5}/b, ^{0.5}/c, d, e\}$ and $\mathit{Increased} = \{b, d\}$, i.e. a is removed from Min. We continue with $b \in Y$ for which $D = \{^{0.5}/a, ^{0.5}/b, ^{0.5}/c, d, e\}$ and $\mathit{Increased} = \{a, d\}$. Thus, b is also removed from Min. Notice that a, b were removed from Min although both the attributes are generators of the upper neighbor D of B. This is correct because neither of them equals $y_B^{\mathrm{C}}(D)$. In the next step, we process $c \in Y$: $D = \{^{0.5}/a, ^{0.5}/b, c, d, e\}$ and $\mathit{Increased} = \{a, b, d\}$. Again, c is removed from Min only this time, c is not even a generator of an upper neighbor of B. Finally, we process $d \in Y$ in which case $D = \{^{0.5}/a, ^{0.5}/b, ^{0.5}/c, d, e\}$ and $\mathit{Increased} = \{a, b\}$. Since $\mathit{Min} = \{d\}$, we add D to $\mathcal{U}$. Then $\mathcal{U}$ is returned as the result of calling Neighbors(B, C).

Now, the algorithm for computing all fixed points can be described as follows. We start with the least fixed point of C which is $\mathrm{C}(\emptyset)$ and add it to the collection of found fixed points. For each newly found fixed point we first use Neighbors from Algorithm 1 to compute its upper neighbors and then we update the information about lower neighbors (D is an upper neighbor of B iff B is a lower neighbor of D). For each upper neighbor which has not been found in previous steps, we recursively repeat the process until we arrive to Y (greatest fixed point of C). The whole procedure is summarized in Algorithm 2.

Algorithm 2 consists of two procedures: Lattice accepts a closure operator $\mathrm{C}\colon L^Y \to L^Y$ and Y as its input and initiates the recursive generation of fixed points starting with the least one. The auxiliary procedure GenerateFrom does the actual job of generating fixed points. Both the procedures use the following variables: $\mathcal{F}$ is a collection of found fixed points, for each $B \in \mathcal{F}$ we denote by B^* the set of all upper neighbors of B, and by B_* we denote the set of all lower neighbors of B. Variable $\mathcal{N}$ is local in GenerateFrom and represents fixed points that were newly found during a particular call of GenerateFrom.

Theorem 4. *Algorithm 2 is correct.*

Proof. Due to the limited scope of the paper, we present only a sketch of the proof (full proof will be presented in the full version of the paper). The crucial observation is that each fixed point of C is in GenerateFrom processed only once. This in ensured at line 10, where GenerateFrom is called only for fixed points that have not been found so far (see definition of $\mathcal{N}$ at line 4). The

	a	b	c	d	e
x_1	1	0.5	0.5	1	1
x_2	1	1	1	1	0.5
x_3	0	0	0.5	0.5	1

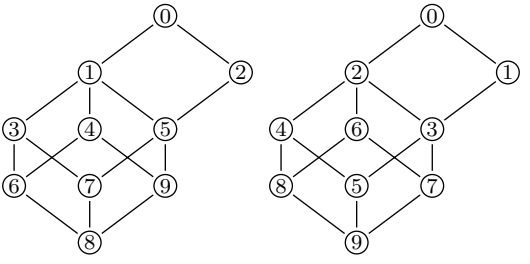


Fig. 1. Input data table and the hierarchy of conceptual clusters

information about lower neighbors (line 6) is also updated correctly because each B (considered only once) is a lower neighbor only of the fixed points which are upper neighbors of B. □

Remark 2. (a) Consider the fuzzy context from Fig.1(left). The corresponding structure of fuzzy concepts computed by Algorithm 2 is depicted in Fig.1 (middle). The numbers in nodes indicate the order in which the intents determining the nodes are computed. Formal concepts corresponding to the nodes of diagram in Fig.1(middle) are the following:

$C_0 : \langle \{x_1, x_2, x_3\}, \{{}^{.5}/c, {}^{.5}/d, {}^{.5}/e\} \rangle$,
$C_1 : \langle \{x_1, x_2, {}^{.5}/x_3\}, \{{}^{.5}/a, {}^{.5}/b, {}^{.5}/c, d, {}^{.5}/e\} \rangle$,
$C_2 : \langle \{x_1, {}^{.5}/x_2, x_3\}, \{{}^{.5}/c, {}^{.5}/d, e\} \rangle$,
$C_3 : \langle \{x_1, x_2\}, \{a, {}^{.5}/b, {}^{.5}/c, d, {}^{.5}/e\} \rangle$,
$C_4 : \langle \{{}^{.5}/x_1, x_2, {}^{.5}/x_3\}, \{{}^{.5}/a, {}^{.5}/b, c, d, {}^{.5}/e\} \rangle$,
$C_5 : \langle \{x_1, {}^{.5}/x_2, {}^{.5}/x_3\}, \{{}^{.5}/a, {}^{.5}/b, {}^{.5}/c, d, e\} \rangle$,
$C_6 : \langle \{{}^{.5}/x_1, x_2\}, \{a, b, c, d, {}^{.5}/e\} \rangle$,
$C_7 : \langle \{x_1, {}^{.5}/x_2\}, \{a, {}^{.5}/b, {}^{.5}/c, d, e\} \rangle$,
$C_8 : \langle \{{}^{.5}/x_1, {}^{.5}/x_2\}, \{a, b, c, d, e\} \rangle$,
$C_9 : \langle \{{}^{.5}/x_1, {}^{.5}/x_2, {}^{.5}/x_3\}, \{{}^{.5}/a, {}^{.5}/b, c, d, e\} \rangle$.

Note that the order in which Algorithm 2 computes the intents (of concepts) differs from the order in which the intents are computed using the algorithm from [3]. The order in which the intents are computed in case of algorithm from [3] is depicted in Fig.1(right).

(b) Both the algorithms introduced in this section use a general fuzzy closure operator C instead of a fixed operator $^{\downarrow\uparrow}$ induced by a data table. The approach via arbitrary C is more general. More importantly, general closure operators play an important role for constraining the output of concept analysis, see [5], for which Algorithm 2 can also be used.

4 Comparison with Other Algorithms and Experiments

Algorithm 2 for computing fixed points of closure operators together with their hierarchy has the same asymptotic complexity as the algorithm proposed in [3]. Taking into account graded attributes, the latter claim can be proved in an analogous way as in [18]. Due to the limited scope of the paper, we omit the proof and, instead, we turn our attention to the practical performance of Algorithm 2 compared to the algorithm from [3].

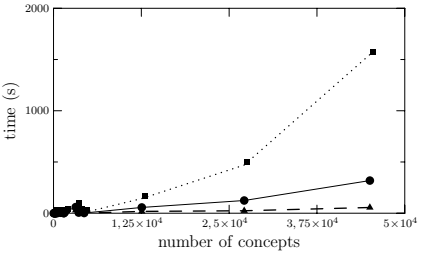


Fig. 2. Running time of algorithms for L with 5 degrees

The efficiency of the implementation of Algorithm 2 depends of the chosen data structures. The representation of $\mathcal{F}$ (and, consequently, B^* and B_* which are likely to be stored along with B) seems to be critical because the elements in $\mathcal{F}$ are frequently accessed (see line 4 of Algorithm 2). To avoid the linear time complexity of accessing elements of $\mathcal{F}$, we have organized $\mathcal{F}$ (i) as a search tree (analogously as in [18]) and (ii) by a dynamic hash table. Moreover, the sets *Min* and *Increased* in Algorithm 1 can be represented by bit arrays which significantly increases the performance (condition at line 7 of Algorithm 1 can be checked by applying the bitwise "and").

We have run several performance tests to compare Algorithm 2 to the algorithm from [3]. Since the algorithm from [3] does not compute the hierarchy of concepts, we have included in our tests an extension of that algorithm which computes the hierarchy after all the concepts are generated. Computing the complete hierarchy has asymptotic time complexity $O(n^2)$, where n is the number of concepts. The algorithms were implemented in ANSI C using the above-mentioned data structures (hashing tables and bit arrays). All experiments were run on otherwise idle Intel Pentium 4 (3.00 GHz CPU, 512 MB RAM).

To compare the performance of the algorithms we did series of experiments with randomly generated data tables with fuzzy attributes. As structures of truth degrees we used finite Łukasiewicz chains of varying size. We were interested in the dependency of running time of the algorithms on the number of generated concepts given by fixed points of $^{\downarrow\uparrow}$. The results of one of the experiments are depicted in Fig. 2. In this particular test we have used a five-element Łukasiewicz chain and we measured the average time needed for computing the concepts and their hierarchy. In the graph, the dashed line with triangles represents the average running time of the algorithm from [3], the dotted line with squares represents the running time of the algorithm from [3] followed by the hierarchy computation, and the solid line with circles corresponds to Algorithm 2.

We can see from the figure that the algorithm from [3] is the best of the three ones if we want to compute the concepts only. If we are interested in generating the concepts along with their hierarchy, Algorithm 2 proposed in this paper is considerably faster than the algorithm from [3] followed by the computation of the hierarchy. Tests with larger data and/or structures of truth degrees have

shown a similar tendency. These experimental results were expected and are in accordance with results presented in [18] for binary data.

References

1. Belohlavek, R.: Fuzzy Relational Systems: Foundations and Principles. Kluwer, Academic/Plenum Publishers, New York (2002)
2. Belohlavek, R.: Fuzzy Galois connections. Math. Logic Quarterly 45(4), 497–504 (1999)
3. Belohlavek, R.: Algorithms for fuzzy concept lattices. In: Proc. Fourth Int. Conf. on Recent Advances in Soft Computing. Nottingham, United Kingdom, pp. 200–205 (December 12–13, 2002)
4. Belohlavek, R.: Concept lattices and order in fuzzy logic. Annals of Pure and Applied Logic 128(1-3), 277–298 (2004)
5. Belohlavek, R., Vychodil, V.: Reducing the size of fuzzy concept lattices by fuzzy closure operators. In: Proc. SCIS & ISIS 2006, pp. 309–314. Tokyo Institute of Technology, Japan, (September 20–24, 2006) ISSN 1880–3741
6. Carpineto, C., Romano, G.: Concept Data Analysis. Theory and Applications. J. Wiley, Chichester (2004)
7. Ganter, B.: Two basic algorithms in concept analysis. FB4-Preprint No. 831, TH Darmstadt (1984)
8. Ganter, B., Wille, R.: Formal concept analysis. Mathematical Foundations. Springer, Heidelberg (1999)
9. Gerla, G.: Fuzzy Logic. Mathematical Tools for Approximate Reasoning. Kluwer, Dordrecht (2001)
10. Goguen, J.A.: The logic of inexact concepts. Synthese 18, 325–373 (1968-69)
11. Gratzer, G.A.: General Lattice Theory. Birkhauser, 2nd edn. (1998)
12. Hájek, P.: Metamathematics of Fuzzy Logic. Kluwer, Dordrecht (1998)
13. Hájek, P.: On very true. Fuzzy Sets and Systems 124, 329–333 (2001)
14. Höhle, U.: On the fundamentals of fuzzy set theory. J. Math. Anal. Appl. 201, 786–826 (1996)
15. Klement, E.P., Mesiar, R., Pap, E.: Triangular Norms. Kluwer, Dordrecht (2000)
16. Klir, G.J., Yuan, B.: Fuzzy Sets and Fuzzy Logic. Theory and Applications. Prentice-Hall, Englewood Cliffs (1995)
17. Kuznetsov, S.O., Obiedkov, S.A.: Comparing performance of algorithms for generating concept lattices. J. Exp. Theor. Artif. Intelligence 14(2/3), 189–216 (2002)
18. Lindig, C.: Fast concept analysis. In: Ganter, B., Mineau, G.W. (eds.) ICCS 2000. LNCS, vol. 1867, pp. 152–161. Springer, Heidelberg (2000)
19. Pollandt, S.: Fuzzy Begriffe. Springer, Heidelberg (1997)
20. Wille, R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In: Rival, I. (ed.) Ordered Sets, pp. 445–470. Reidel, Dordrecht, Boston (1982)
21. Zaki, M.: Mining non-redundant association rules. Data Mining and Knowledge Discovery 9, 223–248 (2004)
22. Zadeh, L.A.: Fuzzy sets. Inf. Control 8(3), 338–353 (1965)

A Version of Lebesgue Decomposition Theorem for Non-additive Measure*

Jun Li[1], Masami Yasuda[2], and Jianzeng Li[3]

[1] School of Science, Communication University of China, Beijing 100024, China
lijun@cuc.edu.cn
[2] Faculty of Science, Chiba University Chiba 263-8522 Japan
yasuda@math.s.chiba-u.ac.jp
[3] School of Information Engineering, Communication University of China, Beijing 100024, China
jzli@cuc.edu.cn

Abstract. In this paper, Lebesgue decomposition type theorems for non-additive measure are shown under the conditions of null-additivity, converse null-additivity, weak null-additivity and σ-null-additivity, etc.. In our discussion, the monotone continuity of set function is not required.

Keywords: Non-additive measure; null-additivity; absolute continuity; Lebesgue decomposition theorem.

1 Introduction

Lebesgue decomposition of a set function μ is stated as: for another given set function ν, μ is represented as $\mu = \mu_c + \mu_s$, where μ_c and μ_s are absolutely continuous and singular with respect to ν, respectively. In measure theory this decomposition is a well-known fact, which is referred to as Lebesgue decomposition theorem [2].

For the case of non-additive measure theory, the situation is not so simple. There are many discussions on Lebesgue decomposition type theorems such as, a version on submeasure (cf. [1]), a version on $\perp$-decomposition measure (cf. [5]), a version on σ-finite fuzzy measure (cf. [3]), and a version on signed fuzzy measure (cf. [10]), and so on. In those discussion, the monotone continuity or autocontinuity of set function are required. However we will try to weaken this condition.

In this paper, we shall show several versions of Lebesgue decomposition theorems of non-additive measure μ for another given non-additive measure ν, where μ is converse null-additive or exhaustive, superadditive or order continuous, and ν is either weakly null-additive and continuous from below or σ-null-additive. In our discussion the monotone continuity of set function is not required, so Lebesgue decomposition theorem is formulated in generality.

* This work was partially supported by the NSFC Grant No.70471063, as well as by the Subject Building Foundation of Communication University of China (No. XK060106).

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 168–173, 2007.

2 Preliminaries

Let X be a non-empty set, $\mathcal{R}$ a σ-ring of subsets of X, and $(X, \mathcal{R})$ denotes the measurable space.

Definition 2.1. A non-additive measure on $\mathcal{R}$ is an real valued set function $\mu : \mathcal{R} \to [0, +\infty)$ satisfying the following two conditions:

(1) $\mu(\emptyset) = 0$;
(2) $A \subset B$ and $A, B \in \mathcal{R} \ \Rightarrow \ \mu(A) \leq \mu(B)$. (monotonicity)

When a non-additive measure μ is continuous from below, it is called a lower semicontinuous fuzzy measure (cf. [8]). In some literature, a set function μ satisfying the conditions (1) and (2) of Definition 2.1 is called a fuzzy measure.

A set function $\mu : \mathcal{R} \to [0, +\infty)$ is said to be (i) *exhaustive* (cf. [6]), if $\lim_{n \to +\infty} \mu(E_n) = 0$ for any infinite disjoint sequence $\{E_n\}_n$; (ii) *order-continuous* (cf. [2]), if $\lim_{n \to \infty} \mu(A_n) = 0$ whenever $A_n \in \mathcal{R}$ and $A_n \searrow \emptyset$ $(n \to \infty)$; (iii) *strongly order-continuous* (cf. [4]), if $\lim_{n \to \infty} \mu(A_n) = 0$ whenever $A_n, A \in \mathcal{R}$, $\searrow A$ and $\mu(A) = 0$; (iv) to have *property* (S) (cf. [7]), if for any $\{A_n\}_{n=1,2,\cdots} \subset \mathcal{R}$ with $\lim_{n \to +\infty} \mu(A_n) = 0$, there exists a subsequence $\{A_{n_i}\}_{i=1,2,\cdots}$ of $\{A_n\}_{n=1,2,\cdots}$ such that $\mu(\limsup A_{n_i}) = 0$.

Definition 2.2. ([9]) Let μ and ν be two non-additive measures. We say that
(1) μ is absolutely continuous of Type I with respect to ν, denoted by $\mu \ll_I \nu$, if $\mu(A) = 0$ whenever $A \in \mathcal{R}$, $\nu(A) = 0$;
(2) μ is absolutely continuous of Type VI with respect to ν, denoted by $\mu \ll_{VI} \nu$, if for any $\epsilon > 0$, there exists $\delta > 0$ such that $\mu(A) < \epsilon$ whenever $A \in \mathcal{R}$, $\nu(A) < \delta$.

Definition 2.3. ([1]) We say that μ is singular with respect to ν, and denote $\mu \perp \nu$, if there is a set $Q \in \mathcal{R}$ such that $\nu(Q) = 0$ and $\mu(E - Q) = 0$ for any $E \in \mathcal{R}$.

For non-additive measures μ and ν, if $\mu \ll_{VI} \nu$, then $\mu \ll_I \nu$. The inverse statement may not be true.

Proposition 2.1. *Let μ and ν be two non-additive measures. If μ is strongly order-continuous, ν have property (S), then $\mu \ll_{VI} \nu$ if and only if $\mu \ll_I \nu$.*

3 Null-Additivity of Set Function

The several kinds of null-additivity of non-additive measure play important role in establishing Lebesgue decomposition type theorems of non-additive measure. We recall them in the following.

Definition 3.1. ([8,9]) A set function $\mu : \mathcal{R} \to [0, +\infty)$ is said to be (i) *null-additive*, if $\mu(E \cup F) = \mu(E)$ whenever $E, F \in \mathcal{R}$ and $\mu(F) = 0$; (ii) *weakly null-additive*, if $\mu(E \cup F) = 0$ whenever $E, F \in \mathcal{R}$, $\mu(E) = \mu(F) = 0$; (iii) *converse*

null-additive, if $\mu(A - B) = 0$ whenever $A \in \mathcal{R},\ B \in A \cap \mathcal{R},\ \mu(B) = \mu(A)$; (iv) *pseudo-null-additive*, if $\mu(B \cup C) = \mu(C)$, whenever $A \in \mathcal{R}, B \in A \cap \mathcal{R}, C \in A \cap \mathcal{R}, \mu(A - B) = \mu(A)$.

Proposition 3.1. *If μ is null-additive, then it is weakly null-additive. If μ is pseudo-null-additive, then it is converse null-additive.*

Definition 3.2. ([6]) A non-additive measure μ is said to be *σ-null-additive*, if for every sequence $\{B_i\}_{i=1,2,\cdots}$ of pairwise disjoint sets from $\mathcal{R}$ and $\mu(B_i) = 0$ we have

$$\mu\left(A \cup \bigcup_{i=1}^{+\infty} B_i\right) = \mu(A) \quad \forall A \in \mathcal{R}.$$

Proposition 3.2. *μ is σ-null-additive if and only if μ is null-additive and $\mu(B_i) = 0$ $(i = 1, 2, \cdots)$ implies $\mu(\cup_{i=k}^{+\infty} B_i) = 0$ for every sequence $\{B_i\}_{i=1,2,\cdots}$ of pairwise disjoint sets from $\mathcal{R}$.*

4 Lebesgue Decomposition Theorems of Non-additive Measure

In this section we show the Lebesgue decomposition type theorems of non-additive measure. Unless stated, in the following we always assume that all set functions are non-additive measure.

Lemma 4.1. *Let μ be converse null-additive non-additive measure on $\mathcal{R}$. Then, there is a set $Q \in \mathcal{R}$ such that $\mu(E - Q) = 0$; and further, if μ is null-additive, then $\mu(E) = \mu(Q \cap E)$ for any $E \in \mathcal{R}$.*

Proof. Let $\alpha = \sup\{\mu(E) \ :\ E \in \mathcal{R}\}$. By the definition of α, we can choose a sequence $\{E_n^{(1)}\}_{n=1,2,\cdots}$ from $\mathcal{R}$ such that for every $n = 1, 2, \cdots$,

$$\alpha - \frac{1}{n} < \mu(E_n^{(1)}) \leq \alpha.$$

Denote $Q_1 = \bigcup_{n=1}^{+\infty} E_n^{(1)}$, then $Q_1 \in \mathcal{R}$. Thus we have

$$\alpha - \frac{1}{n} < \mu(E_n^{(1)}) \leq \mu(Q_1) \leq \alpha,$$

$n = 1, 2, \cdots$. Let $n \to +\infty$, we have $\mu(Q_1) = \alpha$.

Similarly, there exists a sequence $\{E_n^{(2)}\}_{n=1,2,\cdots}$ from $\mathcal{R}$ such that

$$E_n^{(2)} \subset X - Q_1, \quad \forall n \geq 1,$$

and

$$\mu(Q_2) = \sup\{\mu(E) \ :\ E \in \mathcal{R}, \quad E \subset X - Q_1\}$$

where $Q_2 = \bigcup_{n=1}^{+\infty} E_n^{(2)}$.

Let us denote $Q = Q_1 \cup Q_2$. Then $Q \in \mathcal{R}$, and

$$\alpha = \mu(Q_1) \leq \mu(Q) \leq \sup\{\mu(E) \ : \ E \in \mathcal{R}\}.$$

Therefore $\mu(Q) = \mu(Q_1) = \alpha$. By the converse null-additivity of μ, and noting $Q_1 \cap Q_2 = \emptyset$, we have $\mu(Q_2) = \mu(Q - Q_1) = 0$. Therefore, for any $E \in \mathcal{R}$, it is follows from

$$E - Q \subset E - Q_1 \subset X - Q_1$$

that

$$\mu(E - Q) \leq \mu(Q_2) = 0.$$

Thus, for any $E \in \mathcal{R}$, we have $\mu(E - Q) = 0$. When μ is null-additive, then $\mu(E) = \mu((E-Q)\cup(E\cap Q) = \mu(Q\cap E)$ for any $E \in \mathcal{R}$. The proof of the lemma is now complete. □

Lemma 4.2. *Let μ be exhaustive non-additive measure on $\mathcal{R}$. Then, there is a set $Q \in \mathcal{R}$ such that $\mu(E - Q) = 0$; and further, if μ is null-additive, then $\mu(E) = \mu(Q \cap E)$ for any $E \in \mathcal{R}$.*

Proof. It is similar to the proof of Theorem 2 in [3] and Lemma 4.1 above.
The following theorems with their corollaries are several versions of Lebesgue decomposition theorem for non-additive measure.

Theorem 4.1. (Lebesgue decomposition theorem) *Let μ and ν be non-additive measures on $\mathcal{R}$. If μ is either converse null-additive or exhaustive, ν is weakly null-additive and continuous from below, then there exists a set $Q \in \mathcal{R}$ such that those non-additive measures μ_c and μ_s defined by*

$$\mu_c(E) = \mu(E - Q) \quad \textit{and} \quad \mu_s(E) = \mu(E \cap Q), \quad \forall E \in \mathcal{R}$$

satisfy $\mu_c \ll_I \nu$ and $\mu_s \perp \nu$, respectively.

Proof. Put $\mathcal{R}_1 = \{A \in \mathcal{R} \ : \ \nu(A) = 0\}$. Then, $\mathcal{R}_1$ is a σ-subring of $\mathcal{R}$ by the weak null-additivity and continuity from below of ν. By using Lemma 4.1 and 4.2, we may take a set $Q \in \mathcal{R}_1$ such that

$$\mu(Q) = \sup\{\mu(A) \ : \ A \in \mathcal{R}_1\} \quad \text{and} \quad \mu(E - Q) = 0, \quad \forall E \in \mathcal{R}_1.$$

Now we take that $\mu_c(E) = \mu(E - Q)$ and $\mu_s(E) = \mu(E \cap Q)$, $E \in \mathcal{R}$, then μ_c and μ_s satisfy the required properties:

$$\mu_c(E) = 0 \ \text{ if } \ \nu(E) = 0$$

and

$$\mu_s(E - Q) = \mu((E - Q) \cap Q) = \mu(\emptyset) = 0$$

for any $E \in \mathcal{R}$. □

From Proposition 3.1 and Theorem 4.1, we have the following corollary.

Corollary 4.1. *If μ is pseudo-null-additive, ν is null-additive and continuous from below, then there exists a set $Q \in \mathcal{R}$ such that those non-additive measures μ_c and μ_s defined by*

$$\mu_c(E) = \mu(E - Q) \quad \text{and} \quad \mu_s(E) = \mu(E \cap Q), \quad \forall E \in \mathcal{R}$$

satisfy $\mu_c \ll_I \nu$ and $\mu_s \perp \nu$, respectively.

Definition 4.1. ([6]) A set function $\mu : \mathcal{R} \to [0, +\infty)$ is called *superadditive*, if for every $A, B \in \mathcal{R}$ and $A \cap B = \emptyset$,

$$\mu(A \cup B) \geq \mu(A) + \mu(B).$$

Proposition 4.1. *If μ is superadditive, then it is converse null-additive and exhaustive.*

Note that a condition of superadditivity do not imply the order continuity. There is a counterexample: A superadditive non-additive measure is not necessarily order continuous. Counterexample: Let μ be the non-additive measure on 2^N, the power set of the set N of all the positive integers, defined by $\mu(E) = 1$ if E is cofinite (i.e., the complement of a finite set), 0 otherwise. Then obviously μ is superadditive non-additive measure. However $A_n = \{n, n+1, \cdots\}$ for $n = 1, 2, \cdots$ decreasingly converges to the empty set, but the limit of $\{\mu(A_n)\}$ is equal to 1, not zero. This counterexample is noticed by the anonymous referee.

As a direct result of Proposition 4.1 and Theorem 4.1, we can obtain the following theorem immediately.

Theorem 4.2. *Let μ and ν be non-additive measures on $\mathcal{R}$. If μ is superadditive, ν is weakly null-additive and continuous from below, then there exist non-additive measures μ_c and μ_s on $\mathcal{R}$ such that $\mu_c \ll_I \nu$, $\mu_s \perp \nu$, and $\mu \geq \mu_c + \mu_s$.*

By using Proposition 3.2, similar to the proof of Theorem 4.1, we can prove the following theorem.

Theorem 4.3. *Let μ and ν be non-additive measures on $\mathcal{R}$. If μ is either converse null-additive or exhaustive, ν is σ-null-additive, then there exist non-additive measures μ_c and μ_s on $\mathcal{R}$ such that $\mu_c \ll_I \nu$ and $\mu_s \perp \nu$.*

Corollary 4.2. *Let μ and ν be non-additive measures on $\mathcal{R}$. If μ is superadditive, ν is σ-null-additive, then there exist non-additive measures μ_c and μ_s on $\mathcal{R}$ such that $\mu_c \ll_I \nu$, $\mu_s \perp \nu$, and $\mu \geq \mu_c + \mu_s$.*

Note that it is not required that ν has the continuity from below in Theorem 4.3 and Corollary 4.2.

Combining Theorem 4.1 and Proposition 2.1, we can obtain the following result.

Theorem 4.4. *Let μ and ν be non-additive measures on $\mathcal{R}$. If μ is strongly order continuous, ν is weakly null-additive and continuous from below, and have*

property (S), then there exists a set $Q \in \mathcal{R}$ *such that those non-additive measures* μ_c *and* μ_s *defined by*

$$\mu_c(E) = \mu(E - Q) \quad \textit{and} \quad \mu_s(E) = \mu(E \cap Q), \quad \forall E \in \mathcal{R}$$

satisfy $\mu_c \ll_{VI} \nu$ *and* $\mu_s \perp \nu$*, respectively.*

5 Conclusions

There are several versions of Lebesgue decomposition theorem as noted in the first section. Here we proved that Lebesgue decomposition type theorems for non-additive measure are shown under the conditions of null-additivity, converse null-additivity, weak null-additivity and σ-null-additivity. It should be clarified these relations and also considered applications to the various fields.

Acknowledgements. The authors would like to thank anonymous referees a useful comment. We impose the exhaustivity for Lebesgue decomposition theorem of Theorem 4.2 in stead of the order continuity. By this comment, the assertion of Theorem 4.2 is corrected. The research of the first author was supported by the Postdoctoral Research Finance of Communication University of China.

References

1. Dobrakov, I.: On submeasures I. Dissertation Math. 112, 5–35 (1974)
2. Halmos, P.R.: Measure Theory. Van Nostrand, New York (1968)
3. Jiang, Q., Suzuki, H.: Lebesgue and Saks decompositions of σ-finite fuzzy measures. Fuzzy Sets and Systems 75, 373–385 (1995)
4. Li, J.: Order continuous of monotone set function and convergence of measurable functions sequence. Applied Mathematics and Computation 135, 211–218 (2003)
5. Pap, E.: Lebesgue and Saks decompositions of $\perp$ -docomposable measures. Fuzzy Sets and Systems 38, 345–353 (1990)
6. Pap, E.: Null-additive Set Functions. Kluwer, Dordrecht (1995)
7. Sun, Q.: Property(s) of fuzzy measure and Riesz's theorem. Fuzzy Sets and Systems 62, 117–119 (1994)
8. Wang, Z., Klir, G.J.: Fuzzy Measure Theory. Plenum, New York (1992)
9. Wang, Z., Klir, G., Wang, W.: Fuzzy measures defined by fuzzy integral and their absolute continuity. J. Math. Anal. Appl. 203, 150–165 (1996)
10. Zhang, Q., Xu, Y., Du, W.: Lebesgue decompositions theorem for σ-finite signed fuzzy measures. Fuzzy Sets and Systems 101, 445–451 (1999)

Trees in Concept Lattices*

Radim Belohlavek[1,3], Bernard De Baets[2], Jan Outrata[3], and Vilem Vychodil[3]

[1] Dept. Systems Science and Industrial Engineering
T. J. Watson School of Engineering and Applied Science
Binghamton University–SUNY, PO Box 6000, Binghamton, NY 13902–6000, USA
rbelohla@binghamton.edu
[2] Dept. Appl. Math., Biometrics, and Process Control, Ghent University
Coupure links 653, B-9000 Gent, Belgium
bernard.debaets@ugent.be
[3] Dept. Computer Science, Palacky University, Olomouc
Tomkova 40, CZ-779 00 Olomouc, Czech Republic
{jan.outrata, vilem.vychodil}@upol.cz

Abstract. The paper presents theorems characterizing concept lattices which happen to be trees after removing the bottom element. Concept lattices are the clustering/classification systems provided as an output of formal concept analysis. In general, a concept lattice may contain overlapping clusters and need not be a tree. On the other hand, tree-like classification schemes are appealing and are produced by several classification methods as the output. This paper attempts to help establish a bridge between concept lattices and tree-based classification methods. We present results presenting conditions for input data which are sufficient and necessary for the output concept lattice to form a tree after one removes its bottom element. In addition, we present illustrative examples and several remarks on related efforts and future research topics.

1 Introduction

Data tables describing objects and their attributes represent perhaps the most common form of data. Among several methods for analysis of object-attribute data, formal concept analysis (FCA) is becoming increasingly popular, see [7,4]. The main aim of FCA is to extract interesting clusters (called formal concepts) from tabular data along with a partial order of these clusters (called conceptual hierarchy). Formal concepts correspond to maximal rectangles in a data table and are easily interpretable by users. FCA is basically being used two ways. First, as a direct method of data analysis in which case the hierarchically ordered collection of formal concepts extracted from data is presented to a user/expert for further analysis, see e.g. [4] for such examples of FCA applications. Second,

* Supported by Kontakt 1–2006–33 (Bilateral Scientific Cooperation, project "Algebraic, logical and computational aspects of fuzzy relational modelling paradigms"), by grant No. 1ET101370417 of GA AV ČR, by grant No. 201/05/0079 of the Czech Science Foundation, and by institutional support, research plan MSM 6198959214.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 174–184, 2007.

as a data preprocessing method in which case the extracted clusters are used for further processing, see e.g. [13] for applications of FCA in association rules mining.

Unlike several other clustering and classification techniques [1,5], which yield clustering and classification trees, FCA yields diagrams of hierarchically ordered clusters which are richer than trees. Namely, the diagrams are lattices and are called called concept lattices. A practical difference is that concept lattices usually contain overlapping clusters. Another difference is that the clusters in FCA are based on sharing of attributes rather than distance. FCA can thus be thought of as a new method of clustering and classification which is substantially different from conventional methods. Although FCA has been justified by many real-world applications already, see e.g. [4], the following quote from John Hartigan, a leading expert in clustering and classification, is relevant [1,5]:

> "My second remark is about future focus. We pay too much attention to the details of the algorithms. ... It is more important to think about the purposes of clustering, about the types of clusters we wish to construct, ... These details are interesting, ..., but we have plenty of algorithms already. ... what kinds of families of classes should we be looking for? At present, we think of partitions, trees, sometimes overlapping clusters; these structures are a faint echo of the rich classifications available in everyday language. ... We must seek sufficiently rich class of structures ..."

The present paper seeks to contribute to the problem of establishing relationships between FCA and other methods of clustering and classification. Needless to say, this goal requires a long-term effort. In this paper we consider a particular problem. Namely, we present conditions for input data which are sufficient and necessary for the output concept lattice to form a tree after removing its bottom element. In addition, we present illustrative examples and several remarks on related efforts and future research topics. Note that a related problem, namely, of selecting a tree from a concept lattice by means of constraints using attribute-dependency formulas, was considered in [2].

Section 2 presents preliminaries. Section 3 presents the main results, illustrative examples, and remarks. Section 4 presents conclusions and an outline of future research.

2 Preliminaries

In this section, we summarize basic notions of formal concept analysis (FCA). An object-attribute data table describing which objects have which attributes can be identified with a triplet $\langle X, Y, I \rangle$ where X is a non-empty set (of objects), Y is a non-empty set (of attributes), and $I \subseteq X \times Y$ is an (object-attribute) relation. In FCA, $\langle X, Y, I \rangle$ is called a formal context. Objects and attributes correspond to table rows and columns, respectively, and $\langle x, y \rangle \in I$ indicates that object x has attribute y (table entry corresponding to row x and column y contains $\times$ or

1; if $\langle x, y\rangle \notin I$ the table entry contains blank symbol or 0). For each $A \subseteq X$ and $B \subseteq Y$ denote by $A^{\uparrow}$ a subset of Y and by $B^{\downarrow}$ a subset of X defined by

$$A^{\uparrow} = \{y \in Y \mid \text{for each } x \in A\colon \langle x, y\rangle \in I\}, \tag{1}$$

$$B^{\downarrow} = \{x \in X \mid \text{for each } y \in B\colon \langle x, y\rangle \in I\}. \tag{2}$$

Described verbally, $A^{\uparrow}$ is the set of all attributes from Y shared by all objects from A and $B^{\downarrow}$ is the set of all objects from X sharing all attributes from B. A formal concept in $\langle X, Y, I\rangle$ is a pair $\langle A, B\rangle$ of $A \subseteq X$ and $B \subseteq Y$ satisfying $A^{\uparrow} = B$ and $B^{\downarrow} = A$. That is, a formal concept consists of a set A (so-called extent) of objects which fall under the concept and a set B (so-called intent) of attributes which fall under the concept such that A is the set of all objects sharing all attributes from B and, conversely, B is the collection of all attributes from Y shared by all objects from A. Alternatively, formal concepts can be defined as maximal rectangles (submatrices) of $\langle X, Y, I\rangle$ which are full of $\times$'s: For $A \subseteq X$ and $B \subseteq Y$, $\langle A, B\rangle$ is a formal concept in $\langle X, Y, I\rangle$ iff $A \times B \subseteq I$ and there is no $A' \supset A$ or $B' \supset B$ such that $A' \times B \subseteq I$ or $A \times B' \subseteq I$.

A set $\mathcal{B}(X, Y, I) = \{\langle A, B\rangle \mid A^{\uparrow} = B, B^{\downarrow} = A\}$ of all formal concepts in data $\langle X, Y, I\rangle$ can be equipped with a partial order $\leq$ (modeling the subconcept-superconcept hierarchy, e.g. $\mathsf{dog} \leq \mathsf{mammal}$) defined by

$$\langle A_1, B_1\rangle \leq \langle A_2, B_2\rangle \quad \text{iff} \quad A_1 \subseteq A_2 \ (\text{iff } B_2 \subseteq B_1). \tag{3}$$

Under $\leq$, $\mathcal{B}(X, Y, I)$ happens to be a complete lattice, called a concept lattice of $\langle X, Y, I\rangle$, the basic structure of which is described by a so-called main theorem of concept lattices [7]:

Theorem 1 (Main Theorem of Concept Lattices). (1) *The set $\mathcal{B}(X, Y, I)$ is under $\leq$ a complete lattice where the infima and suprema are given by*

$$\textstyle\bigwedge_{j\in J}\langle A_j, B_j\rangle = \big\langle \bigcap_{j\in J} A_j, (\bigcup_{j\in J} B_j)^{\downarrow\uparrow}\big\rangle, \tag{4}$$

$$\textstyle\bigvee_{j\in J}\langle A_j, B_j\rangle = \big\langle (\bigcup_{j\in J} A_j)^{\uparrow\downarrow}, \bigcap_{j\in J} B_j\big\rangle. \tag{5}$$

(2) *Moreover, an arbitrary complete lattice $\mathbf{V} = \langle V, \leq\rangle$ is isomorphic to $\mathcal{B}(X, Y, I)$ iff there are mappings $\gamma\colon X \to V$, $\mu\colon Y \to V$ such that*

(i) *$\gamma(X)$ is $\bigvee$-dense in V, $\mu(Y)$ is $\bigwedge$-dense in V;*

(ii) *$\gamma(x) \leq \mu(y)$ iff $\langle x, y\rangle \in I$.* □

Note that a subset $K \subseteq V$ is $\bigvee$-dense in V ($\bigwedge$-dense in V) if for every $v \in V$ there is $K' \subseteq K$ such that $v = \bigvee K'$ ($v = \bigwedge K'$). Note also that operators $^{\uparrow}$ and $^{\downarrow}$ form a so-called Galois connection [7] and that $\mathcal{B}(X, Y, I)$ is in fact a set of all fixed points of $^{\uparrow}$ and $^{\downarrow}$. That is, $^{\uparrow}$ and $^{\downarrow}$ satisfy the following conditions:

$$A \subseteq A^{\uparrow\downarrow}, \tag{6}$$

$$\text{if } A_1 \subseteq A_2 \text{ then } A_2{}^{\uparrow} \subseteq A_1{}^{\uparrow}, \tag{7}$$

$$B \subseteq B^{\downarrow\uparrow}, \tag{8}$$

$$\text{if } B_1 \subseteq B_2 \text{ then } B_2{}^{\downarrow} \subseteq B_1{}^{\downarrow}, \tag{9}$$

for each $A, A_1, A_2 \subseteq X$ and $B, B_1, B_2 \subseteq Y$. Furthermore, the composed operators $^{\uparrow\downarrow} : 2^X \to 2^X$ and $^{\downarrow\uparrow} : 2^Y \to 2^Y$ are closure operators in X and Y,

respectively. As a consequence, $A \subseteq X$ is an extent of some concept in $\mathcal{B}(X, Y, I)$ (i.e., there is $B \subseteq Y$ such that $\langle A, B\rangle \in \mathcal{B}(X, Y, I)$) iff $A = A^{\uparrow\downarrow}$, i.e. A is closed under $^{\uparrow\downarrow}$. Analogously for intents.

Concept lattices are the primary output of formal concept analysis. There is another output of FCA which is equally important, namely, so-called non-redundant bases of attribute implications. An attribute implication is an expression $A \Rightarrow B$ where $A, B \subseteq Y$ with Y being the same set of attributes as above. An attribute implication $A \Rightarrow B$ is called true in $M \subseteq Y$, written $M \models A \Rightarrow B$, if the following condition is satisfied:

$$\text{if } A \subseteq M \text{ then } B \subseteq M.$$

If $M \subseteq Y$ represents a set of attributes of some object x then $M \models A \Rightarrow B$ has the following meaning: "if x has all attributes from A, then x has all attributes from B". Thus, attribute implications are particular if-then rules describing dependencies between attributes.

Given a formal context $\langle X, Y, I\rangle$, for each $x \in X$ we define a set I_x of attributes $I_x = \{y \in Y \mid \langle x, y\rangle \in I\}$, i.e. I_x is the set of all attributes of object x in $\langle X, Y, I\rangle$. Notice that I_x corresponds to a row in data table representing format context $\langle X, Y, I\rangle$. An attribute implication $A \Rightarrow B$ is called true in $\langle X, Y, I\rangle$, written $I \models A \Rightarrow B$, iff $I_x \models A \Rightarrow B$ for each $(x \in X)$. Hence, $I \models A \Rightarrow B$ iff for each object $x \in X$ we have: if x has all attributes from A, then x has all attributes from B.

The set of all attribute implications which are true in $\langle X, Y, I\rangle$ is, along with the concept lattice $\mathcal{B}(X, Y, I)$, the basic output of FCA. Unfortunately, the set of all attribute implications is usually too large and it cannot be presented to users directly. Therefore, we use special indirect description of all attribute implications being true in $\langle X, Y, I\rangle$. Namely, we select from all the attribute implications in question a small subset from which the other implications follow. This can be done using the following notions.

Let T be any set of attribute implications. A set $M \subseteq Y$ of attributes is called a model of T, if $M \models A \Rightarrow B$ for each $A \Rightarrow B \in T$. The set of all models of T will be denoted by $\mathrm{Mod}(T)$, i.e.

$$\mathrm{Mod}(T) = \{M \subseteq Y \mid \text{for each } A \Rightarrow B \in T\text{: } M \models A \Rightarrow B\}. \tag{10}$$

An attribute implication $A \Rightarrow B$ follows from T ($A \Rightarrow B$ is semantically entailed by T), written $T \models A \Rightarrow B$, if $M \models A \Rightarrow B$ for each $M \in \mathrm{Mod}(T)$. A set T of attribute implications is called complete in $\langle X, Y, I\rangle$ if, for each attribute implication $A \Rightarrow B$, we have

$$T \models A \Rightarrow B \quad \text{iff} \quad I \models A \Rightarrow B,$$

i.e., if the attribute implications which are entailed by T are exactly the attribute implications which are true in $\langle X, Y, I\rangle$. Hence, if T is complete in $\langle X, Y, I\rangle$, then T describes exactly the attribute implications which are true in $\langle X, Y, I\rangle$. This is important especially if T is "reasonably small". Therefore, we define the following notion. A set T of attribute implications is a non-redundant basis of $\langle X, Y, I\rangle$ if (i) T is complete in $\langle X, Y, I\rangle$ and (ii) no proper subset of T is

complete in $\langle X, Y, I \rangle$. Alternatively, a non-redundant basis of $\langle X, Y, I \rangle$ can be described as complete sets of attribute implications such that no implication in the set is entailed by the other implications in that set. There have been proposed algorithms to generate, given $\langle X, Y, I \rangle$, a non-redundant basis of $\langle X, Y, I \rangle$, see e.g. [6,7,10].

For a detailed information on formal concept analysis and lattice theory we refer to [4,7,8] where a reader can find theoretical foundations, methods and algorithms, and applications in various areas.

3 Trees in Concept Lattices

In this section we will be interested in concept lattices corresponding to trees. Trees are usually defined as undirected graphs which are acyclic and connected [9]. Since we are going to identify trees in particular ordered sets, we deal with trees as with ordered sets. In particular, a finite partially ordered set $\langle U, \leq \rangle$ will be called a tree if for each $a, b \in U$:

(i) there is a supremum of a and b in $\langle U, \leq \rangle$, and
(ii) there is an infimum of a and b in $\langle U, \leq \rangle$ iff a and b are comparable (i.e., iff $a \leq b$ or $b \leq a$).

Obviously, $\langle U, \leq \rangle$ being a tree corresponds to the usual graph-theoretical representation of a rooted tree. The root of $\langle U, \leq \rangle$ is the supremum of all elements from U (which exists in $\langle U, \leq \rangle$ because U is finite). An element $u \in U$ is a direct descendant of $w \in U$ iff $u < w$, and there is no $v \in U$ such that $u < v < w$.

From Theorem 1 it follows that each concept lattice is a complete lattice. Hence, the above-mentioned condition (i) is satisfied for each $\mathcal{B}(X, Y, I)$. On the other hand, (ii) need not be satisfied. It is easily seen that (ii) is satisfied iff $\mathcal{B}(X, Y, I)$ is linearly ordered. So, the whole concept lattice is a tree iff it is linear, which is not a worthwhile observation because linear trees are a degenerate form of trees and therefore not interesting. Because of the observation we have just made, we turn our attention to trees which form important parts of concept lattices. We focus mainly on trees which appear in $\mathcal{B}(X, Y, I)$ if we remove its least element.

Since concept lattices are complete lattices, each concept lattice $\mathcal{B}(X, Y, I)$ has both the greatest and least element. Namely, $\langle X, X^{\uparrow} \rangle$ is the greatest element (concept of all objects) of $\mathcal{B}(X, Y, I)$ and $\langle Y^{\downarrow}, Y \rangle$ is the least one (concept of objects sharing all attributes from Y). If $\langle X, Y, I \rangle$ does not contain an attribute shared by all objects (i.e., a table representing $\langle X, Y, I \rangle$ does not contain a column full of $\times$'s), which is quite common if $\langle X, Y, I \rangle$ represents a real-world data, then $\langle X, X^{\uparrow} \rangle$ equals $\langle X, \emptyset \rangle$. Analogously, there is no object sharing all the attributes from Y (i.e., a table representing $\langle X, Y, I \rangle$ does not contain a row full of $\times$'s), $\langle Y^{\downarrow}, Y \rangle$ becomes $\langle \emptyset, Y \rangle$.

In what follows we investigate under which conditions $\mathcal{B}(X, Y, I)$ becomes a tree if we remove its least element.

3.1 Formal Contexts Generating Trees

For brevity, let $\mathcal{B}(X,Y,I)-\{\langle Y^{\downarrow},Y\rangle\}$ be denoted by $\mathcal{B}^{\curlywedge}(X,Y,I)$. Note that if we consider $\mathcal{B}^{\curlywedge}(X,Y,I)$, we assume that it is equipped with a partial order which is a restriction of the partial order defined by (3) to elements of $\mathcal{B}^{\curlywedge}(X,Y,I)$.

The following assertion characterizes when $\mathcal{B}^{\curlywedge}(X,Y,I)$ is a tree in terms of extents of formal concepts.

Theorem 2. *Let $\langle X,Y,I\rangle$ be a formal context. Then $\mathcal{B}^{\curlywedge}(X,Y,I)$ is a tree iff, for each concepts $\langle A,B\rangle,\langle C,D\rangle\in\mathcal{B}(X,Y,I)$ at least one of the following is true:*

(i) $A\subseteq C$ *or* $C\subseteq A$,
(ii) $A\cap C\subseteq Y^{\downarrow}$.

Proof. Let $\mathcal{B}^{\curlywedge}(X,Y,I)$ be a tree. Take any concepts $\langle A,B\rangle,\langle C,D\rangle\in\mathcal{B}(X,Y,I)$. If (i) is satisfied for A and C, we are done. Hence, assume that (i) is not satisfied, i.e. we have $A\not\subseteq C$ and $C\not\subseteq A$. From the definition of $\leq$, see (3), it follows that $\langle A,B\rangle\not\leq\langle C,D\rangle$ and $\langle C,D\rangle\not\leq\langle A,B\rangle$, i.e. formal concepts $\langle A,B\rangle$ and $\langle C,D\rangle$ are incomparable. Therefore, both $\langle A,B\rangle$ and $\langle C,D\rangle$ are in $\mathcal{B}^{\curlywedge}(X,Y,I)$. Since $\mathcal{B}^{\curlywedge}(X,Y,I)$ is supposed to be a tree, infimum of $\langle A,B\rangle$ and $\langle C,D\rangle$ does not exist in $\mathcal{B}^{\curlywedge}(X,Y,I)$. It means that the infimum of $\langle A,B\rangle$ and $\langle C,D\rangle$ in $\mathcal{B}(X,Y,I)$ is $\langle Y^{\downarrow},Y\rangle$ because $\mathcal{B}^{\curlywedge}(X,Y,I)$ results from $\mathcal{B}(X,Y,I)$ by removing $\langle Y^{\downarrow},Y\rangle$ and $\mathcal{B}(X,Y,I)$ is a complete lattice. Using (4), we get that $Y^{\downarrow}=A\cap C$, showing (ii).

Conversely, let (i) and (ii) be satisfied for any $\langle A,B\rangle,\langle C,D\rangle\in\mathcal{B}(X,Y,I)$. Take $\langle A,B\rangle,\langle C,D\rangle\in\mathcal{B}^{\curlywedge}(X,Y,I)$ such that $\langle A,B\rangle$ and $\langle C,D\rangle$ are incomparable. Such $\langle A,B\rangle$ and $\langle C,D\rangle$ cannot satisfy (i), i.e. we have $A\cap C\subseteq Y^{\downarrow}$. Hence, using (4), the infimum of $\langle A,B\rangle$ and $\langle C,D\rangle$ in $\mathcal{B}(X,Y,I)$ is the least element of $\mathcal{B}(X,Y,I)$. As a consequence, $\langle A,B\rangle$ and $\langle C,D\rangle$ does not have an infimum in $\mathcal{B}^{\curlywedge}(X,Y,I)$ which proves that $\mathcal{B}^{\curlywedge}(X,Y,I)$ is a tree. □

Theorem 2 can also be formulated in terms of intents of formal concepts:

Corollary 1. *$\mathcal{B}^{\curlywedge}(X,Y,I)$ is a tree iff, for each $\langle A,B\rangle,\langle C,D\rangle\in\mathcal{B}(X,Y,I)$ we either have* (i) $B\subseteq D$ *or* $D\subseteq B$, *or* (ii) $(B\cup D)^{\downarrow}\subseteq Y^{\downarrow}$. □

Remark 1. If $\langle Y^{\downarrow},Y\rangle$ is equal to $\langle\emptyset,Y\rangle$, i.e. if the table representing $\langle Y^{\downarrow},Y\rangle$ does not contain a row full of $\times$'s (or 1's), then (iii) in Theorem 2 simplifies to $A\cap C=\emptyset$, i.e. A and C are required to be disjoint.

Example 1. Consider a set of objects $X=\{1,2,\ldots,14\}$ (objects are denoted by numbers) and a set of attributes $Y=\{g,h,\ldots,z\}$. If we consider a formal context $\langle X,Y,I\rangle$ which is represented by the data table in Fig. 1(left) then the corresponding $\mathcal{B}^{\curlywedge}(X,Y,I)$, which is depicted in Fig. 1(right), is a tree. The root of the tree represents concept $\langle X,\emptyset\rangle$. The other nodes are numbered and the intents of the corresponding concepts are the following:

1: $\{i,r\}$,
2: $\{i,o,r\}$,
3: $\{i,l,o,r\}$,
4: $\{i,r,w\}$,
5: $\{m,s,z\}$,
6: $\{g,m,n,s,v,z\}$,
7: $\{g,j,m,n,p,s,t,v,z\}$,
8: $\{g,j,k,m,n,p,s,t,u,v,z\}$,
9: $\{g,m,n,q,s,v,z\}$,
10: $\{m,s,x,z\}$,
11: $\{h,m,s,x,z\}$,
12: $\{m,s,y,z\}$.

	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
1.	1	0	0	0	0	0	1	1	0	0	1	0	1	0	0	1	0	0	0	1
2.	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1
3.	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0
4.	1	0	0	1	0	0	1	1	0	1	0	0	1	1	0	1	0	0	0	1
5.	1	0	0	1	1	0	1	1	0	1	0	0	1	1	1	1	0	0	0	1
6.	1	0	0	0	0	0	1	1	0	0	0	0	1	0	0	1	0	0	0	1
7.	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	1
8.	0	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0
9.	1	0	0	1	0	0	1	1	0	1	0	0	1	1	0	1	0	0	0	1
10.	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0
11.	0	1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	1
12.	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0
13.	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	1
14.	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

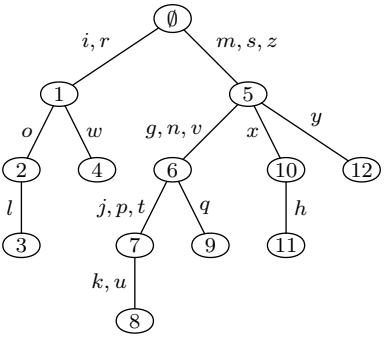


Fig. 1. Formal context generating a tree

If two nodes are connected by an edge, the lower concept has a strictly greater intent. Using this observation, we can decorate edges of the tree by attributes being added to intents of lower concepts as it is shown in Fig. 1 (right). On can check that all the intents from $\mathcal{B}(X, Y, I)$ satisfy condition of Theorem 1.

Now, an important question is, if we can check that $\mathcal{B}^{\curlywedge}(X, Y, I)$ is a tree directly from the context $\langle X, Y, I\rangle$, i.e. without computing the set of all concepts first. We shall show that this is indeed possible. We will take advantage of the following notion.

Definition 1. *Let $\langle X, Y, I\rangle$ be a formal context. We say that $\langle X, Y, I\rangle$ generates a tree if $\mathcal{B}^{\curlywedge}(X, Y, I)$ is a tree.*

Recall that due to (2), $\{y\}^{\downarrow}$ is a set of all objects sharing the attribute y. That is, $\{y\}^{\downarrow}$ naturally corresponds to a column in data table representing $\langle X, Y, I\rangle$. Such "columns" will play an important role in the following theorem which characterizes contexts generating trees.

Theorem 3. *Let $\langle X, Y, I\rangle$ be a formal context. Then $\langle X, Y, I\rangle$ generates a tree iff, for any attributes $y_1, y_2 \in Y$, at least one of the following conditions is true:*

(i) $\{y_1\}^{\downarrow} \subseteq \{y_2\}^{\downarrow}$,
(ii) $\{y_2\}^{\downarrow} \subseteq \{y_1\}^{\downarrow}$,
(iii) $\{y_1\}^{\downarrow} \cap \{y_2\}^{\downarrow} \subseteq Y^{\downarrow}$.

Proof. Assume that $\langle X, Y, I\rangle$ generates a tree, i.e. $\mathcal{B}^{\curlywedge}(X, Y, I)$ is a tree. Each pair of the form $\langle \{y\}^{\downarrow}, \{y\}^{\downarrow\uparrow}\rangle$ is a formal concept from $\mathcal{B}(X, Y, I)$, see [7]. Therefore, Theorem 2 yields that the above conditions (i)–(iii), being particular instances of (i) and (ii) from Theorem 2, are satisfied.

Conversely, suppose that $\langle X, Y, I\rangle$ does not generate a tree. Thus, there are some incomparable formal concepts $\langle A_1, B_1\rangle, \langle A_2, B_2\rangle \in \mathcal{B}(X, Y, I)$ whose infimum is not equal to the least element of $\mathcal{B}(X, Y, I)$. That is, for $\langle A_1, B_1\rangle$ and

$\langle A_2, B_2\rangle$ we have $A_1 \not\subseteq A_2$, $A_2 \not\subseteq A_1$, and $A_1 \cap A_2 \not\subseteq Y^{\downarrow}$. Note that as a consequence we get that $B_1 \not\subseteq B_2$ and $B_2 \not\subseteq B_1$. We now show that we can pick from B_1 and B_2 two attributes violating the above conditions (i)–(iii). Since $B_1 \not\subseteq B_2$, there is $y_1 \in B_1$ such that $y_1 \notin B_2$. Analogously, there is $y_2 \in B_2$ such that $y_2 \notin B_1$ because $B_2 \not\subseteq B_1$. For y_1 and y_2 we can show that (i) is not satisfied. Indeed, from $y_1 \in B_1 = A_1{}^{\uparrow}$ and (9) it follows that

$$A_1 \subseteq \{y_1\}^{\downarrow}. \tag{11}$$

Moreover, $y_2 \notin B_1 = A_1{}^{\uparrow}$ gives that there is $x \in A_1$ such that $\langle x, y_2\rangle \notin I$. Hence, there is $x \in A_1$ such that $x \notin \{y_2\}^{\downarrow}$, i.e. we get

$$A_1 \not\subseteq \{y_2\}^{\downarrow}. \tag{12}$$

As an immediate consequence of (11) and (12) we get that $\{y_1\}^{\downarrow} \not\subseteq \{y_2\}^{\downarrow}$, i.e. condition (i) is violated. In a symmetric way (i.e., with y_1 and y_2 interchanged), we can also show that (ii) is violated. So, now it remains to show that (iii) cannot be satisfied. But this is now easy to see. From $y_1 \in B_1$, $y_2 \in B_2$, and (9) we get $A_1 = B_1{}^{\downarrow} \subseteq \{y_1\}^{\downarrow}$ and $A_2 = B_2{}^{\downarrow} \subseteq \{y_2\}^{\downarrow}$ which yield $A_1 \cap A_2 \subseteq \{y_1\}^{\downarrow} \cap \{y_2\}^{\downarrow}$. Therefore, from $A_1 \cap A_2 \not\subseteq Y^{\downarrow}$ it follows that $\{y_1\}^{\downarrow} \cap \{y_2\}^{\downarrow} \not\subseteq Y^{\downarrow}$, showing that (iii) is not satisfied. Altogether, we have shown that if $\langle X, Y, I\rangle$ does not generate a tree, then there are $y_1, y_2 \in Y$ such that neither of (i)–(iii) is satisfied. □

Remark 2. Conditions (i)–(iii) from Theorem 3 say that, roughly speaking, for each two columns of a data table, either one of the columns is contained in the other, or the columns have in common only attributes shared by all objects. In particular, if no row of the data table contains all ×'s (or 1's), the latter condition says that the columns do not have any attributes in common. Note that (i)–(iii) can be checked with asymptotic time complexity $O(n^3)$, where n is the maximum of $|X|$ and $|Y|$.

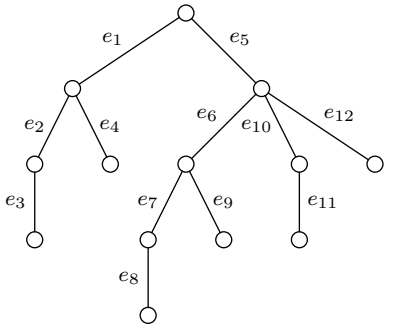
Theorem 3 can be restated as follows:

Corollary 2. *A formal context $\langle X, Y, I\rangle$ generates a tree iff, for any $y_1, y_2 \in Y$, we either have $\{y_1\}^{\downarrow} \cap \{y_2\}^{\downarrow} \in \{\{y_1\}^{\downarrow}, \{y_2\}^{\downarrow}\}$, or $\{y_1\}^{\downarrow} \cap \{y_2\}^{\downarrow} \subseteq Y^{\downarrow}$.* □

We now turn our attention to a converse problem. Given a tree (defined possibly by its graph-theoretical representation), we wish to find a formal context which generates the tree. First, let us note that for each tree such a context exists. This is, in fact, a consequence of the main theorem of concept lattices. In a more detail, consider a graph $\mathbf{G} = \langle V, E\rangle$ which is a tree [9]. We say that edge $e_1 \in E$ in under $e_2 \in E$ (in $\mathbf{G}$) if $\mathbf{G}$ contains a path $v_1, e_1, \ldots, v_2, e_2, \ldots$ ending in the root node of $\mathbf{G}$ (for the notions involved, see [9]). We now get the following characterization.

Theorem 4. *Let $\mathbf{G} = \langle V, E\rangle$ be a tree. Define a formal context $\langle E, E, I_{\mathbf{G}}\rangle$ such that $\langle e_1, e_2\rangle \in I_{\mathbf{G}}$ iff e_1 is under e_2 in $\mathbf{G}$. Then $\langle E, E, I_{\mathbf{G}}\rangle$ generates a tree which is isomorphic to $\mathbf{G} = \langle V, E\rangle$.*

Proof. Follows from Theorem 1. We omit details of the proof due to the limited scope of this paper. □



	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}	e_{11}	e_{12}
e_1	1	0	0	0	0	0	0	0	0	0	0	0
e_2	1	1	0	0	0	0	0	0	0	0	0	0
e_3	1	1	1	0	0	0	0	0	0	0	0	0
e_4	1	0	0	1	0	0	0	0	0	0	0	0
e_5	0	0	0	0	1	0	0	0	0	0	0	0
e_6	0	0	0	0	1	1	0	0	0	0	0	0
e_7	0	0	0	0	1	1	1	0	0	0	0	0
e_8	0	0	0	0	1	1	1	1	0	0	0	0
e_9	0	0	0	0	1	1	0	0	1	0	0	0
e_{10}	0	0	0	0	1	0	0	0	0	1	0	0
e_{11}	0	0	0	0	1	0	0	0	0	1	1	0
e_{12}	0	0	0	0	1	0	0	0	0	0	0	1

Fig. 2. Tree and its generating formal context

Example 2. If we return to Example 1 and consider the tree from Fig. 1 (right) as an input tree, we may construct a formal context generating that tree as follows. First, we choose a labeling of edges. For instance, we may choose labeling as in Fig. 2 (left). Then, a formal context which corresponds to $\langle E, E, I_{\mathbf{G}} \rangle$ from Theorem 4 is given by data table in Fig. 2 (right). Since we have labeled the edges in a depth-first manner, $\langle E, E, I_{\mathbf{G}} \rangle$ is in a lower-triangular form. By Theorem 4, tree $\mathcal{B}^{\curlywedge}(E, E, I_{\mathbf{G}})$ generated from $\langle E, E, I_{\mathbf{G}} \rangle$ is isomorphic to the initial tree.

3.2 Characterization of Trees by Attribute Implications

In the previous section, we have shown that contexts generating trees can be characterized based on the dependencies between attributes (columns of data tables representing formal contexts). Since attribute dependencies are often expressed by attribute implications, it is tempting to look at trees in a concept lattice from the point of view of attribute implications.

The following assertion characterizes contexts generating trees by means of attribute implications.

Theorem 5. *Let $\langle X, Y, I \rangle$ be a formal context. Then $\langle X, Y, I \rangle$ generates a tree iff, for any attributes $y_1, y_2 \in Y$, at least one of the following is true:*

(i) $I \models \{y_1\} \Rightarrow \{y_2\}$,
(ii) $I \models \{y_2\} \Rightarrow \{y_1\}$,
(iii) $I \models \{y_1, y_2\} \Rightarrow Y$.

Proof. Note that attribute implications being true in $\langle X, Y, I \rangle$ can be characterized using operators $^{\uparrow}$ and $^{\downarrow}$ induced by $\langle X, Y, I \rangle$. Namely, one can check that $I \models A \Rightarrow B$ iff, for each $x \in X$, if $A \subseteq \{x\}^{\uparrow}$ then $B \subseteq \{x\}^{\uparrow}$ which is iff, for each $x \in X$, if $x \in A^{\downarrow}$ then $x \in B^{\downarrow}$ which is true iff $A^{\downarrow} \subseteq B^{\downarrow}$, see [7]. Thus, (i) and (ii) are true iff $\{y_1\}^{\downarrow} \subseteq \{y_2\}^{\downarrow}$ and $\{y_2\}^{\downarrow} \subseteq \{y_1\}^{\downarrow}$, cf. Theorem 3 (i) and (ii). Moreover, (iii) is true iff we have $\{y_1\}^{\downarrow} \cap \{y_2\}^{\downarrow} = (\{y_1\} \cup \{y_2\})^{\downarrow} = \{y_1, y_2\}^{\downarrow} \subseteq Y^{\downarrow}$. Using Theorem 3, we finally obtain that $\langle X, Y, I \rangle$ generates a tree iff, for any $y_1, y_2 \in Y$, at least one of (i)–(iii) is true. □

3.3 Algorithms For Trees in Concept Lattices

Trees in concept lattices, as they were introduced in previous sections, can be computed by algorithms for computing formal concepts. Currently, there have been proposed several algorithms, see e.g. [4,7,12] and a survey paper [11]. Some of the algorithms for FCA simplify in case of contexts generating trees.

For instance, Lindig's algorithm for generating formal concepts simplifies due to the fact that it is no longer necessary to organize found concepts in some type of searching structure, because we cannot generate the same concept multiple times. Indeed, recall from [12] that Lindig's algorithm is based on the NextNeighbors procedure which, given a concept as its input, generates all its (lower or upper) neighbors. Then, all concepts are computed using a recursive procedure which fist uses NextNeighbors to compute neighbors of a given concept and then recursively processes all the neighbors to obtain further concepts. During the computation, the original procedure has to ensure that no concept will be computed twice (or multiple times). Therefore, the procedure must organize all found concepts in a suitable searching structure which allows us to check whether a concept has already been found. Needless to say, the searching structure should be efficient because the tests of presence of a concept between the found concepts influences the overall efficiency of the procedure. The searching structure is usually implemented as a searching tree or a hashing table.

In case of context generating trees, this part of the algorithm need not be implemented at all because the only concept that can be computed multiple times is $\langle Y^{\downarrow}, Y \rangle$ which is excluded from $\mathcal{B}^{\curlywedge}(X, Y, I)$. This allows to have an algorithm which is faster and simpler to implement.

4 Conclusions and Future Research

We presented conditions for input data for FCA which are sufficient and necessary for the output concept lattice to form a tree after one removes its bottom element. Trees are the most common structures which appear in traditional clustering and classification. Out long-term effort will be focused on establishing connections between FCA and other clustering and classification methods. First, establishing such relationships helps us see the pros and cons, and limits of the respective methods. Second, with the basic relationships established, one can hopefully enrich the respective methods by techniques used in the other methods. The problems we want to address next include the following ones:

- A concept lattice can be seen as consisting of several overlapping trees. What can we say about such a "decomposition" of a concept lattice into trees? What are the relationships between these trees?
- A user of FCA might be interested in a part of a concept lattice rather than in the whole lattice. Particularly, that part might be a tree, but other parts might be interesting as well. The issue of selecting parts of concept lattices by constraints was discussed in [2,3]. In particular, it can be shown that a tree contained in a concept lattice can be selected by means of a particular

closure operator. Constraints which lead to tree-like parts of concept lattices need to be further investigated.
– Investigate connections between concept lattices and decision trees. Both concept lattices and decision trees contain clusters of objects in their nodes. Leafs of a decision tree correspond to particular attribute-concepts. A construction of a decision tree may be thought of as a selection of a particular part from a concept lattice. Containment of decision trees in concept lattices needs to be further investigated.

References

1. Arabie, P., Hubert, L.J., DeSoete, G.: Clustering and Classification. World Scientific (1996)
2. Belohlavek, R., Sklenar, V.: Formal concept analysis constrained by attribute-dependency formulas. In: Ganter, B., Godin, R. (eds.) ICFCA 2005. LNCS (LNAI), vol. 3403, pp. 176–191. Springer, Heidelberg (2005)
3. Belohlavek, R., Vychodil, V.: Formal concept analysis with constraints by closure operators. In: Schärfe, H., Hitzler, P., Øhrstrøm, P. (eds.) ICCS 2006. LNCS (LNAI), vol. 4068, pp. 131–143. Springer, Heidelberg (2006)
4. Carpineto, C., Romano, G.: Concept Data Analysis. Theory and Applications. J. Wiley, Chichester (2004)
5. Everitt, B.S., Landau, S., Leese, M.: Cluster Analysis, 4th edn. Oxford University Press, New York (2001)
6. Ganter, B.: Begriffe und Implikationen, manuscript (1998)
7. Ganter, B., Wille, R.: Formal Concept Analysis. Mathematical Foundations. Springer, Heidelberg (1999)
8. Gratzer, G.A.: General Lattice Theory. 2nd edn. Birkhauser (1998)
9. Grimaldi, R.P.: Discrete and Combinatorial Mathematics, Pearson Education, 5th edn. (2002)
10. Guigues, J.-L., Duquenne, V.: Familles minimales d'implications informatives resultant d'un tableau de données binaires. Math. Sci. Humaines 95, 5–18 (1986)
11. Kuznetsov, S.O., Obiedkov, S.A.: Comparing performance of algorithms for generating concept lattices. J. Exp. Theor. Artif. Intelligence 14(2/3), 189–216 (2002)
12. Lindig, C.: Fast concept analysis. In: Ganter, B., Mineau, G.W. (eds.) ICCS 2000. LNCS, vol. 1867, pp. 152–161. Springer, Heidelberg (2000)
13. Zaki, M.J.: Mining non-redundant association rules. Data Mining and Knowledge Discovery 9, 223–248 (2004)

An Axiomatization of Shapley Values of Games on Set Systems

Aoi Honda[1] and Yoshiaki Okazaki[2]

[1] Kyushu Institute of Technology, 680-2 Kawazu, Iizuka, Fukuoka 820-8502, Japan
aoi@ces.kyutech.ac.jp
http://www.ces.kyutech.ac.jp/~aoi/
[2] Kyushu Institute of Technology, 680-2 Kawazu, Iizuka, Fukuoka, 820-8502, Japan
okazaki@ces.kyutech.ac.jp

Abstract. An axiomatization of a generalized Shapley value of games is proposed. The authors follow Faigle and Kern, in the sense that our basic material is the maximal chains of the underlying set system. This generalized Shapley value may have applicability to the game on set systems which satisfy the condition of a sort of normality.

1 Introduction

Let $X = \{1, 2, \ldots, n\}$ be a set consisted of n players. Then a subset of X is called a coalition. A function which shows the profit by any coalition $v : 2^X \to I\!R$ is called a cooperative game. The solution of the game is a function from the whole set of the game to n dimension real vector which measures each player's contribution or share-out. The Shapley value and the Banzhaf value are known as the solution, and they are characterized by natural axiomatizations [3] [4]. In this paper we shall show the axiomatization of the generalized Shapley value. Faigle and Kern have generalized the Shapley value using the concept of the maximal chain [5], it can be applied to the multi-choice game. Algaba et al. have also generalized it using the concept of the interior which can be applied to the game defined on antimatroid set systems, and they have given its axiomatizations [1]. Using these generalized Shapley value, we can obtain solutions of the bi-capacity and the multi-choice game and so on. We shall make Faigle and Kern's Shapley value more general and also show its axiomatization.

2 Set System and Shapley Value of a Game on It

We begin by introducing some notations and definitions. Throughout this paper, we consider a finite universal set $X = \{1, 2, \ldots, n\}$, $n \geq 1$, and 2^X denotes the power set of X. Let us consider $\mathfrak{S}$ a subset of 2^X which contains $\emptyset$ and X. Then we call $(X, \mathfrak{S})$ (or simply $\mathfrak{S}$ if no confusion occurs) a *set system*. A set system endowed with inclusion is a particular case of a *partially ordered set* $(\mathfrak{S}, \subseteq)$, i.e., a set $\mathfrak{S}$ endowed with a partial order (reflexive, antisymmetric and transitive) as $\subseteq$.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 185–192, 2007.

Let $A, B \in \mathfrak{S}$. We say that A is *covered* by B, and write $A \prec B$ or $B \succ A$, if $A \subsetneq B$ and $A \subseteq C \subsetneq B$ together with $C \in \mathfrak{S}$ imply $C = A$.

Definition 1 (maximal chain of set system). *Let $\mathfrak{S}$ be a set system. We call $\mathscr{C}$ a* maximal chain of $\mathfrak{S}$ *if $\mathscr{C} = (C_0, C_1, \ldots, C_m)$ satisfies $\emptyset = C_0 \prec C_1 \prec \cdots \prec C_m = X, C_i \in \mathfrak{S}, i = 0, \ldots, m$.*

The length of the maximal chain $\mathscr{C} = (C_0, C_1, \ldots, C_m)$ is m. We denote the set of all m-length maximal chains of $\mathfrak{S}$ by $\Gamma_m(\mathfrak{S})$, $1 \leq m \leq n$.

Example 1. Let $X := \{1, 2, 3\}, \mathfrak{S}_1 := (\{\emptyset, \{1\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, X\}$. Then the maximal chains of $\mathfrak{S}_1$ are $\mathscr{C}_1 := \{\emptyset, \{1\}, \{1, 2\}, X\}, \mathscr{C}_2 := \{\emptyset, \{1\}, \{1, 3\}, X\}$, $\mathscr{C}_3 := \{\emptyset, \{2, 3\}, X\}, \Gamma_2(\mathfrak{S}) = \{\mathscr{C}_3\}$ and $\Gamma_3(\mathfrak{S}) = \{\mathscr{C}_1, \mathscr{C}_2\}$ (Fig. 1).

Remark 1. $(X, 2^X)$ has $n!$ maximal chains and all of their length are n.

Definition 2 (totally ordered set system). *We say that $(X, \mathfrak{S})$ is a* totally ordered set system *if for any $A, B \in \mathfrak{S}$, either $A \subseteq B$ or $A \supsetneq B$.*

If $(X, \mathfrak{S})$ is a totally ordered set system, then it has only one maximal chain which length is n.

Definition 3 (normal set system). *We say that $(X, \mathfrak{S})$ is a* normal set system *if for any $A \in \mathfrak{S}$ there exists $\mathscr{C} \in \Gamma_n(\mathfrak{S})$ satisfying $A \in \mathscr{C}$.*

Example 2. $(\{1, 2, 3\}, \{\emptyset, \{1\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\})$ is not a normal set system, because there are not any 3-length maximal chains which includes $\{2, 3\}$ (Fig. 1).

Remark 2. Normality does not mean that all length of maximal chains are n. In fact, $(\{1, 2, 3, 4\}, \{\emptyset, \{1\}, \{3\}, \{1, 2\}, \{3, 4\}, \{1, 2, 3\}, \{2, 3, 4\}, \{1, 2, 3, 4\}\})$, which has 3-length maximal chain $(\emptyset, \{3\}, \{1, 2, 3\}, \{1, 2, 3, 4\})$, is a normal set system (Fig. 2).

Definition 4 (game on a set system). *Let $(X, \mathfrak{S})$ be a set system. A function $v : \mathfrak{S} \to \mathbb{R}$ is a* game on $(X, \mathfrak{S})$ *if it satisfies $v(\emptyset) = 0$.*

Definition 5 (Shapley value of game on $(X, 2^X)$[3]). *Let v be a game on $(X, 2^X)$. The* Shapley value of v, $\Phi(v) = (\phi^1(v), \ldots, \phi^n(v)) \in [0, 1]^n$ *is defined by*

$$\phi^i(v) := \sum_{A \subseteq X \setminus \{i\}} \gamma^n_{|A|}(v(A \cup \{i\}) - v(A)), \ i = 1, \ldots, n,$$

where

$$\gamma^n_k := \frac{(n - k - 1)!\ k!}{n!}.$$

Remark that $\sum_{i=1}^n \phi^i(v) = v(X)$ holds. The Shapley value can be represented by using the maximal chains as follows.

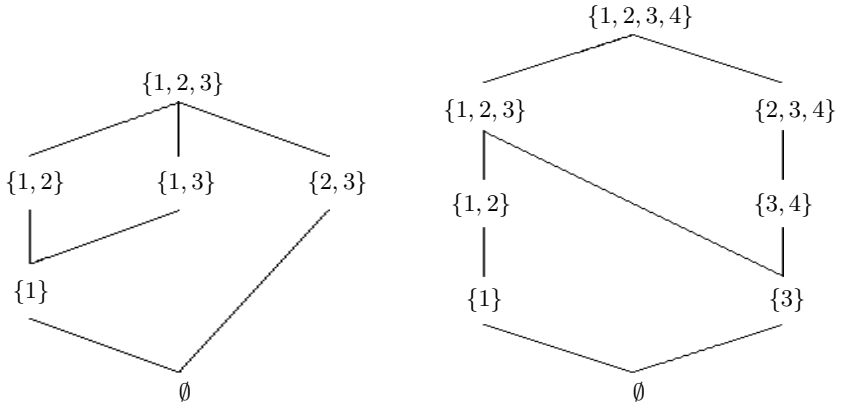


Fig. 1. set systems

Proposition 1. *Fix arbitrarily $i \in X$. For any $\mathscr{C} \in \Gamma_n(2^X)$, there exists a unique $A_{\mathscr{C}} \in \mathscr{C}$ such that $i \notin A_{\mathscr{C}}$ and $A_{\mathscr{C}} \cup \{i\} \in \mathscr{C}$, and*

$$\phi^i(v) = \frac{1}{n!} \sum_{\mathscr{C} \in \Gamma_n(2^X)} (v(A_{\mathscr{C}} \cup \{i\}) - v(A_{\mathscr{C}}))$$

holds.

The fact is well known in the game theory. We give a proof of Proposition 1 for the sake of completeness.

Proof. $|\Gamma_n(2^X)| = n!$ holds. First, we show that for any $\mathscr{C} \in \Gamma_n(2^X)$, there is an $A_{\mathscr{C}} \in \mathscr{C}$ such that $i \notin A_{\mathscr{C}}$ and $A_{\mathscr{C}} \cup \{i\} \in \mathscr{C}$. Fix $\mathscr{C} = (C_0, C_1, \ldots, C_m) \in \Gamma_n(2^X)$. We have for $k = 1, \ldots, m$, $|C_k \setminus C_{k-1}| = 1$ so that $m = n$ holds. We have $C_k \setminus C_{k-1} \neq C_j \setminus C_{j-1}$ for $k < j$ because if $C_k \setminus C_{k-1} = C_j \setminus C_{j-1}$ then $C_j \supseteq C_k \setminus C_{k-1}$. But since $k < j$, $C_{j-1} \supseteq C_k$ therefore $C_{j-1} \supseteq C_k \setminus C_{k-1}$ which is a contradiction. Hence for any $i \in X$, there is an $A_{\mathscr{C}} \in \mathscr{C}$, which satisfies $i \notin A_{\mathscr{C}}$ and $A_{\mathscr{C}} \cup \{i\} \in \mathscr{C}$.

Next we show that for $A \subseteq X \setminus \{i\}$, the number of chains which include $A \cup \{i\}$ and A is $(n - |A| - 1)!|A|!$. Fix arbitrarily $i \in X$. Number of chains from $A \cup \{i\}$ to X is $(n - |A| - 1)!$ and chains from $\emptyset$ to A is $|A|!$. Hence the number of chains which include $A \cup \{i\}$ and A is $|A|! \cdot (n - |A| - 1)!$. Therefore

$$\frac{1}{n!} \sum_{\mathscr{C} \in \Gamma_n(2^X)} (v(A_{\mathscr{C}} \cup \{i\}) - v(A_{\mathscr{C}})) = \frac{(n - |A| - 1)! \cdot |A|!}{n!} \sum_{A \in X \setminus \{i\}} (v(A \cup \{i\}) - v(A)),$$

which completes the proof.

Faigle and Kern had generalized the Shapley value for applying the multi-choice game using the concept of the maximal chain. We extend their Shapley value for applying to more general cases.

Definition 6 (Shapley value of game on set system). *Let v be a game on a normal set system $(X, \mathfrak{S})$. The Shapley value of v, $\Phi(v) = (\phi^1(v), \ldots, \phi^n(v)) \in \mathbb{R}^n$ is defined by*

$$(\mathbf{FK}) \qquad \phi^i_{\mathrm{FK}}(v) := \frac{1}{|\Gamma_n(\mathfrak{S})|} \sum_{\mathscr{C} \in \Gamma_n(\mathfrak{S})} (v(A_{\mathscr{C}} \cup \{i\}) - v(A_{\mathscr{C}})),$$

where $A_{\mathscr{C}} := A \in \mathscr{C} \in \Gamma_n(\mathfrak{S})$ such that $i \notin A$ and $A \cup \{i\} \in \mathscr{C}$.

We discuss the domains of Φ. Let $(X, \mathfrak{S})$ be a normal set system and let v be a game on $(X, \mathfrak{S})$. Then we call $(X, \mathfrak{S}, v)$ a game space. Let Σ_n be the set of all normal set systems of $X := \{1, 2, \ldots, n\}$ and let $\Delta_{\mathfrak{S}}$ be the set of all game spaces defined on a normal set system $(X, \mathfrak{S})$. The domain of Φ is $\Delta := \bigcup_{n=1}^{\infty} \bigcup_{\mathfrak{S} \in \Sigma_n} \Delta_{\mathfrak{S}}$, and Φ is a function defined on Δ to $\mathbb{R}^n$. We denote simply $\Phi(v)$ and $\phi^i(v)$ instead of $\Phi(X, \mathfrak{S}, v)$ and $\phi^i(v)$ as far as no confusion occurs.

We introduce further concepts about games, which will be useful for stating axioms.

Definition 7 (dual game). *Let v be a game on $(X, \mathfrak{S})$. Then the* dual game *of v is defined on $\mathfrak{S}^d := \{A \in 2^X \mid A^c \in \mathfrak{S}\}$ by $v^d(A) := 1 - v(A^c)$ for any $A \in \mathfrak{S}^d$, where $A^c := X \setminus A$.*

Definition 8 (permutation of v). *Let v be a game on $(X, \mathfrak{S})$ and π be a permutation on X. Then the* permutation *of v by π is defined on $\pi(\mathfrak{S}) := \{\pi(A) \in 2^X \mid A \in \mathfrak{S}\}$ by $\pi \circ v(A) := v(\pi^{-1}(A))$.*

Let us consider a chain of length 2 as a set system, denoted by $\mathbf{2}$ (e.g., $\{\emptyset, \{1\}, \{1, 2\}\}$), and a game $v^{\mathbf{2}}$ on it. We denote by the triplet $(0, u, t)$ the values of $v^{\mathbf{2}}$ along the chain. We suppose $\mathbf{2} := \{\emptyset, \{1\}, \{1, 2\}\}$ unless otherwise noted.

Definition 9 (embedding of $v^{\mathbf{2}}$). *Let v be a game on a totally ordered normal set system $(X, \mathfrak{S})$, where $\mathfrak{S} := \{C_0, \ldots, C_n\}$ such that $C_{i-1} \prec C_i$, $i = 1, \ldots, n$, and let $v^{\mathbf{2}} := (0, u, t), t \neq 0$, be a game on $\mathbf{2}$. Then for $C_k \in \mathfrak{S}$, v^{C_k} is called the* embedding *of $v^{\mathbf{2}}$ into v at C_k and defined on the totally ordered normal set system $(X^{C_k}, \mathfrak{S}^{C_k})$ by*

$$v^{C_k}(A) := \begin{cases} v(C_j), & \text{if } A = C_j, j < k, \\ v(C_{k-1}) + \frac{u}{t} \cdot \big(v(C_k) - v(C_{k-1})\big), & \text{if } A = C'_k, \\ v(C_{j-1}), & \text{if } A = C'_j, j > k, \end{cases} \tag{1}$$

where $\{i_k\} := C_k \setminus C_{k-1}, i'_k \neq i''_k$, $(X \setminus \{i_k\}) \cap \{i'_k, i''_k\} = \emptyset, X^{C_k} := (X \setminus \{i_k\}) \cup \{i'_k, i''_k\}, C'_k := (C_k \setminus \{i_k\}) \cup \{i'_k\}, C'_j := (C_{j-1} \setminus \{i_k\}) \cup \{i'_k, i''_k\}$ for $j > k$, and $\mathfrak{S}^{C_k} := \{C_0, \ldots, C_{k-1}, C'_k, C'_{k+1}, \ldots, C'_{n+1}\}$.

Remark that more properly, the dual game of v is the dual game space of the game space $(X, \mathfrak{S}, v)$ which is defined by $(X, \mathfrak{S}, v)^d := (X, \mathfrak{S}^d, v^d)$ with $\mathfrak{S}^d := \{A^c \in 2^X \mid A \in \mathfrak{S}\}$, the permutation of v is the permutation of the game space $(X, \mathfrak{S}, v)$ which is defined by $(X, \mathfrak{S}, v)^\pi := (X, \pi(\mathfrak{S}), \pi \circ v)$, and the embedding of $(0, u, t), t \neq 0$, into v is the embedding of the game space $(\{1, 2\}, \mathbf{2}, (0, u, t))$ into the game space $(X, \mathfrak{S}, v)$, and it is defined by $(X, \mathfrak{S}, v)^{C_k} := (X^{C_k}, \mathfrak{S}^{C_k}, v^{C_k})$.

3 Axiomatization of the Shapley Value of Games

We introduce six axioms for the Shapley value of games on normal set systems.

Axiom 1 (continuity of $v^{\mathbf{2}}$). *The function $\phi^1(0, u, t)$ is continuous with respect to u.*

Axiom 2 (efficiency of $v^{\mathbf{2}}$). *For any game $v^{\mathbf{2}} = (0, u, t)$ on $\mathbf{2}$, $\phi^1(0, u, t) + \phi^2(0, u, t) = t = v(X)$ holds.*

Axiom 3 (dual invariance of $v^{\mathbf{2}}$). *For any game $v^{\mathbf{2}} = (0, u, t)$ on $\mathbf{2}$, $\Phi(0, u, t) = \Phi(0, u, t)^d$ holds.*

Axiom 4 (embedding efficiency). *Let $(X, \mathfrak{S})$ be a totally ordered normal set system and let $\mathfrak{S} := \{C_0, \ldots, C_n\}, C_{i-1} \prec C_i, i = 1, \ldots, n$. Then for any v on $(X, \mathfrak{S})$, any $(0, u, t), t \neq 0$, and any $C_k \in \mathfrak{S}$, $\phi^i(v^{C_k}) = \phi^i(v)$ for any $i \neq i'_k, i''_k$, $\phi^{i'_k}(v^{C_k}) = \phi^{i_k}(v) \cdot \phi^1(0, u, t)/t$ and $\phi^{i''_k}(v^{C_k}) = \phi^{i_k}(v) \cdot \phi^2(0, u, t)/t$ hold, where $\{i_k\} := C_k \setminus C_{k-1}$.*

Axiom 5 (convexity). *Let $(X, \mathfrak{S})$, $(X, \mathfrak{S}_1)$ and $(X, \mathfrak{S}_2)$ be normal set systems satisfying $\Gamma_n(\mathfrak{S}_1) \cup \Gamma_n(\mathfrak{S}_2) = \Gamma_n(\mathfrak{S})$ and $\Gamma_n(\mathfrak{S}_1) \cap \Gamma_n(\mathfrak{S}_2) = \emptyset$ and v be a game on $\mathfrak{S}$. Then there exists $\alpha \in]0, 1[$ satisfying that for every game v on $\mathfrak{S}$ and for every $i \in X$, $\phi^i(v) = \alpha\phi^i(v|_{\mathfrak{S}_1}) + (1 - \alpha)\phi^i(v|_{\mathfrak{S}_2})$.*

Axiom 6 (permutation invariance). *Let $(X, \mathfrak{S})$ be a normal set system and v be a game on $(X, \mathfrak{S})$. Then for any permutation π on X satisfying $\pi(\mathfrak{S}) = \mathfrak{S}$, $\phi^i(v) = \phi^{\pi(i)}(\pi \circ v), i = 1, \ldots, n$ holds.*
Then we obtain the following theorem.

Theorem 1. *Let v be a game on a normal set system $(X, \mathfrak{S})$. Then there exists a unique function satisfying* Axioms 1, 2, 3, 4, 5 *and* 6, *and it is given by* (FK).

Now, we discuss in detail the above axioms.

3.1 Efficiency of $v^{\mathbf{2}}$

More generally, for any game on a normal set system, Axiom 2 holds.

Proposition 2. *Let $(X, \mathfrak{S})$ be a normal set system. Then for any game on $(X, \mathfrak{S})$, $\sum_{i=1}^n \phi^i_{\mathrm{FK}}(v) = v(X)$ holds.*

3.2 Dual Invariance

More generally, for any game on a normal set system, $\Phi(v)$ is dual invariant.

Proposition 3. *Let $(X, \mathfrak{S})$ be a normal set system. Then for any v on $(X, \mathfrak{S})$, $\Phi_{\mathrm{FK}}(v^d) = \Phi_{\mathrm{FK}}(v)$.*

3.3 Embedding Efficiency

Let v be a game on a totally ordered normal set system $\mathfrak{S} := \{C_0, \ldots, C_n\}$ such that $C_{i-1} \prec C_i$, $i = 1, \ldots, n$, Then the embedding at C_k into v by $(0, u, t)$ means that $i_k := C_k \backslash C_{k-1}$ is splitted to $\{i'_k, i''_k\}$. Axiom 4 implies $\phi^{i'_k}(v^{C_k}) + \phi^{i''_k}(v^{C_k}) = \phi^{i_k}(v)$ and $\phi^i(v^{C_k}) = \phi^i(v)$ for $i \neq i', i''$, so that Axiom 4 is natural property in the meaning of the contributions of i'_k and i''_k.

4 Application to Game on Lattice

The *lattice* $(L, \leq)$ is a partially ordered set such that for any pair $x, y \in L$, there exist a least upper bound $x \vee y$ (supremum) and a greatest lower bound $x \wedge y$ (infimum) in L. Consequently, for finite lattices, there always exist a greatest element (supremum of all elements) and a least element (infimum of all elements), denoted by $\top, \bot$ (see [2]). Our approach may have applicability to games defined on lattices which satisfy a kind of normality by the translation from lattices to set systems (cf. [6]).

Definition 10 (game on lattice). *Let $(L, \leq)$ be a finite lattice with least element denoted by $\bot$. A* game *on L is a function $v : L \to \mathbb{R}$ satisfying $v(\bot) = 0$.*

Evidently the set system is not necessarily a lattice. Moreover, the normal set system is not necessarily a lattice. Indeed, take $X := \{1, 2, 3, 4\}$ and $\mathfrak{S} := \{\emptyset, \{1\}, \{4\}, \{1, 2\}, \{1.3\}, \{2, 4\}, \{3, 4\}, \{1, 2, 4\}, \{1, 3, 4\}, X\}$. Then, $(X, \mathfrak{S})$ is a normal set system, but it is not a lattice, because there is not the supremum of $\{1\}$ and $\{4\}$ (Fig. 2).

Definition 11 (join-irreducible element). *An element $x \in (L, \leq)$ is* join-irreducible *if for all $a, b \in L$, $x \neq \bot$ and $x = a \vee b$ implies $x = a$ or $x = b$.*

We denote the set of all join-irreducible elements of L by $\mathcal{J}(L)$.

The mapping η for any $a \in L$, defined by

$$\eta(a) := \{x \in \mathcal{J}(L) \mid x \leq a\}$$

is a lattice-isomorphism of L onto $\eta(L) := \{\eta(a) \mid a \in L\}$, that is, $(L, \leq) \cong (\eta(L), \subseteq)$. (Fig. 3)

Translating lattices which is underlying space of a game v to set systems, we obtain a set of players as $\mathcal{J}(L)$ and a set system as $\eta(L)$ and we can apply Definition 6 to a game on them if that is the case where the set system is normal.

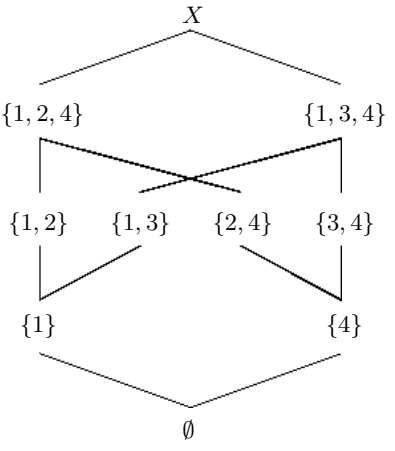


Fig. 2.

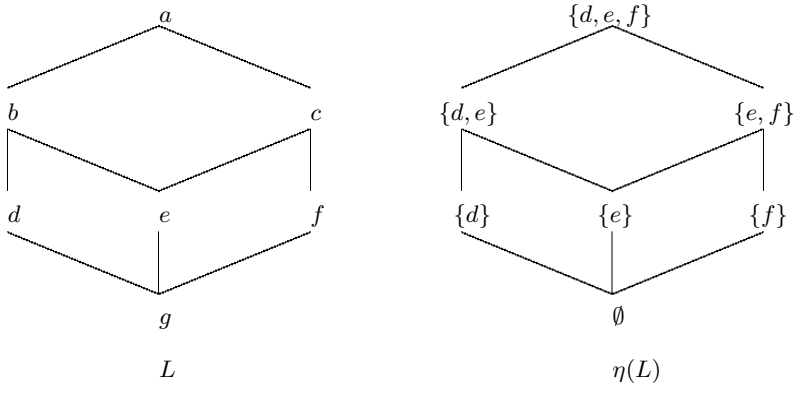


Fig. 3. Translation of lattice

In this paper we treat games defined on normal set systems. If the underlying space is not normal, Definition 6 can not be applied to such games and this fact is natural. Because $\phi^i(v)$ means a sort of the contribution of player i and is calculated as an average of i's contributions $v(A \cup \{i\}) - v(A)$. For instance, let $X := \{1, 2, 3\}$ and $\mathfrak{S}_1 := \{\emptyset, \{1\}, \{1, 2, 3\}\}$ which is not normal, and let v be a game on $(X, \mathfrak{S}_1)$. Then we cannot know contributions of each single $\{2\}$ nor $\{3\}$. If we regard $\{\emptyset, \{1\}, \{1, 2, 3\}\}$ as the lattice, not the set system, the situation is a little different. In this case, the name of elements are just the label. We have $\mathcal{J}(\mathfrak{S}_1) = \{\{1\}, \{1, 2, 3\}\}$ and $|\mathcal{J}(\mathfrak{S}_1)| = 2$, so that considering $\mathfrak{S}_1$ as a set system $(\mathcal{J}(\mathfrak{S}_1), \mathfrak{S}_1)$, we can apply Definition 6 to the game on $\mathfrak{S}_1$, and we obtain $\phi^{\{1\}}(v)$ and $\phi^{\{1,2,3\}}(v)$.

References

1. Algaba, E., Bilbao, J.M., van den Brink, R., Jime' nez-Losada, A.: Axiomatizations of the Shapley value for cooperative games on antimatroids. Math. Meth. Oper. Res. 57, 49–65 (2003)
2. Davey, B.A., Priestley, H.A.: Introduction to lattices and order. Cambridge University Press, Cambridge (1990)
3. Shapley, L.: A value for n-person games. In: Kuhn, H., Tucker, A. (eds.) Contributions to the Theory of Games, Vol. II Annals of Mathematics Studies, vol. 28, pp. 307–317. Princeton University Press, Princeton, NJ (1953)
4. Banzhaf III, J.F.: Weighted voting doesn't work. A mathematical analysis. Rutgers Law Review 19, 313–345 (1965)
5. Faigle, U., Kern, W.: The Shapley value for cooperative games under precedence constraints. Int. J. of Game Theory 21, 249–266 (1992)
6. Honda, A., Grabisch, M.: Entropy of capacities on lattices. Information Sciences 176, 3472–3489 (2006)

Formulation of Fuzzy c-Means Clustering Using Calculus of Variations

Sadaaki Miyamoto

Department of Risk Engineering
Faculty of Systems and Information Engineering
University of Tsukuba, Ibaraki 305-8573, Japan
miyamoto@risk.tsukuba.ac.jp

Abstract. A membership matrix of fuzzy c-mans clustering is associated with the corresponding fuzzy classification rules as membership functions defined on the whole space. In this paper such functions in fuzzy c-means and possibilistic clustering are directly derived using the calculus of variations. Consequently, the present formulation generalizes the ordinary fuzzy c-means and moreover related methods can be discussed within this framework.

1 Introduction

The method of fuzzy c-means clustering is now used in a variety of applications and one of the best-known techniques for data analysis. Nevertheless, there are still problems to be solved and theoretical properties to be uncovered.

One of such problems is the derivation of a fuzzy classification rule also called a fuzzy classifier. Although there are different proposals for fuzzy classifiers, a standard classification rule is the membership function that interpolates the membership values of the objects for clustering, whereby many theoretical properties of fuzzy c-means can be uncovered [13]. Although a standard fuzzy classifier is associated with each formulation of fuzzy c-means, it cannot be derived directly. In this paper we derive a classifier using the calculation of variations [6]. This method derives the membership matrix and the classifier at the same time by changing the distribution of objects. We also have similar results on possibilistic clustering [11]. Moreover generalizations of the existing formulation of fuzzy c-means and possibilistic clustering are obtained using the present formulation of the calculus of variations.

The outline of this paper is as follows. After reviewing briefly fuzzy clustering algorithms and the concept of classification functions which is obtained from the membership matrices in Section 2, we propose the formulation of the clustering problem based on the calculus of variations, whereby classification functions are directly derived in Section 3. Section 4 discusses a generalization of the formulation which includes ordinary membership matrices, noise clustering, and other variations of the fuzzy c-means clustering. Moreover, Section 5 is concerned with theoretical properties of the classification functions derived in Section 3.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 193–203, 2007.

2 Different Objective Functions

Objects for clustering are denoted by $x_k = (x^1, \ldots, x_k^p) \in \boldsymbol{R}^p$, $k = 1, \ldots, n$, a vector in the p-dimensional Euclidean space. Cluster centers are $v_i = (v_i^1, \ldots, v_i^p)^T$, $i = 1, \ldots, c$, where c is the number of clusters. An abbreviated symbol $V = (v_1, \ldots, v_c)$ is used for the whole collection of cluster centers. The matrix $U = (u_{ik})$, $(i = 1, \ldots, c,\ k = 1, \ldots, n)$ is used as usual, where u_{ik} means the degree of belongingness of object x_k to cluster i. In fuzzy c-means clustering, the constraint for a fuzzy partition is

$$M_f = \{U = (u_{ik}) : \sum_{i=1}^{c} u_{ik} = 1, \forall k; u_{jk} > 0, \forall j, k\}.$$

The dissimilarity for clustering is the standard squared Euclidean distance between an individual and a cluster center:

$$D_{ik} = \|x_k - v_i\|^2.$$

We moreover use the dissimilarity between a generic element $x \in \boldsymbol{R}^p$ and v_i, which is denoted by

$$D(x, v_i) = \|x - v_i\|^2.$$

2.1 Fuzzy c-Means and Possibilistic Clustering

As is well-known, fuzzy c-means clustering is based on the optimization of an objective function. We consider three different types of objective functions.

$$J_1(U, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ik})^m D_{ik} \tag{1}$$

$$J_2(U, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} \{u_{ik} D_{ik} + \nu u_{ik} \log u_{ik}/e\} \tag{2}$$

$$J_3(U, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ik})^m D_{ik} + \sum_{i=1}^{c} \eta_i \sum_{k=1}^{n} (1 - u_{ik})^m \tag{3}$$

J_1 is the well-known objective function proposed by Bezdek [1] and Dunn [5], while J_2 is an entropy-based method proposed by several authors (e.g. [12]). J_3 is the function for the possibilistic clustering [11].

In the next description of the alternative optimization **FCM** of fuzzy c-means clustering, we can use $J = J_i$, $i = 1, 2$ and $M = M_f$.

Algorithm FCM
FCM0. Set an initial value $\bar{V}$.
FCM1. Find optimal solution of J with respect to U while V is fixed: put

$$\bar{U} = \arg \min_{U \in M} J(U, \bar{V}).$$

FCM2. Find optimal solution of J with respect to V while U is fixed: put

$$\bar{V} = \arg\min_{V} J(\bar{U}, V).$$

FCM3. If the solution $(\bar{U}, \bar{V})$ is convergent, stop; else go to FCM1.
End of FCM.

We show solutions of each step and we write, for simplicity, u_{ik} instead of $\bar{u}_{ik}$, v_i instead of $\bar{v}_i$, without confusion. Moreover the solution u_{ik} when J_ℓ $(\ell = 1, \cdots, 3)$ is used is denoted by $u_{ik}^{(\ell)}$, if necessary. However, the superscript (ℓ) is omitted when no confusion arises. For this purpose Let us define three basic functions $F^{(\ell)}(x; v_i)$ $(\ell = 1, \ldots, 3)$ for the respective objective functions.

$$F^{(1)}(x; v_i) = \frac{1}{\|x - v_i\|^{\frac{2}{m-1}}},$$
$$F^{(2)}(x; v_i) = \exp(-\nu^{-1}\|x - v_i\|^2),$$
$$F^{(3)}(x; v_i) = \frac{1}{1 + (\|x - v_i\|^2/\eta_i)^{\frac{1}{m-1}}}.$$

Remark that $u_{ik}^{(\ell)}$ $(\ell = 1, 2)$ are represented as

$$u_{ik}^{(\ell)} = \frac{F^{(\ell)}(x_k; v_i)}{\sum_{j=1}^{c} F^{(\ell)}(x_k; v_j)}. \tag{4}$$

Notice that J_3 cannot be used for fuzzy c-means in general, but for the specific case of $m = 2$ in J_3, equation (4) can be used for $u_{ik}^{(3)}$.

On the other hand, solutions of cluster centers are as follows.

$$v_i = \frac{\sum_{k=1}^{n} (u_{ik})^m x_k}{\sum_{k=1}^{n} (u_{ik})^m}. \tag{5}$$

where $m = 1$ for J_2.

Possibilistic clustering. In possibilistic clustering, the constraint M_f is not imposed on U. Instead of M_f, we assume

$$M = (\boldsymbol{R}^+)^{cn}, \tag{6}$$

that is, practically no constraint for U. When J_1 is assumed, the solution $\bar{U}$ is trivial, and hence this objective function is useless. Thus we have

$$u_{ik}^{(\ell)} = F^{(\ell)}(x_k; v_i), \quad \ell = 2, 3. \tag{7}$$

for J_2 and J_3.

The solution for the cluster center is given by the same equation (5).

We also note J_2 and J_3 with $m = 2$ are applicable to both fuzzy c-means and possibilistic clustering [4,12].

Notice that the function $F^{(\ell)}(x; v_i)$ is directly employed to solutions u_{ik} in possibilistic clustering by substituting $x = x_k$. whereas the solution of fuzzy c-means is the ratio of this function $F^{(\ell)}(x; v_i)$ to the sum of $F^{(\ell)}(x; v_j)$ for all cluster j.

2.2 Fuzzy Classification Functions

Let use define three functions of $x \in \boldsymbol{R}^p$:

$$\mathbf{u}_i^{(\ell)}(x; V) = \frac{F^{(\ell)}(x; v_i)}{\sum_{j=1}^{c} F^{(\ell)}(x; v_j)}, \quad i = 1, 2, \ldots, c \tag{8}$$

where $m = 2$ for $\ell = 3$. This function implies a fuzzy classification rule or fuzzy classifier associated with the membership matrix $(u_{ik}^{(\ell)})$, as it is a function $\mathbf{u}_i^{(\ell)}(\cdot; V)$: $\boldsymbol{R}^p \to [0, 1]$ and interpolates the membership value: $\mathbf{u}_i^{(\ell)}(x_k; V) = u_{ik}^{(\ell)}$ with $x = x_k$. The function moreover has the parameter V. We hereafter call this function as a fuzzy classification function to distinguish it from the term of fuzzy classifier, as the latter term is used for different functions of classification rules [2].

The fuzzy classification function can immediately be obtained by replacing the object symbol x_k by the variable x. This derivation is, however, artificial and does not uncover a fundamental idea behind the fuzzy classification rules.

Note that such fuzzy classification functions have been used for other purposes than fuzzy c-means. For example fuzzy self-organizing scheme [14] and supervised classification rules [9] employ the fuzzy classification function. A fuzzy classification function is thus important in itself apart from the alternative optimization algorithm. Hence direct derivation of a fuzzy classification function without the concept of clustering should be considered.

Note 1. In the case of possibilistic clustering, it is immediate to see that $F(x; v_i)^{(\ell)}$ $(\ell = 2, 3)$ is the fuzzy classification function.

3 Direct Derivation of Classification Functions

For this purpose we first notice that the classification rules should be determined using prototypes $v_1, \cdots, v_c$. In clustering these are cluster centers but they are not necessarily centers but may be other prototypes in the case of supervised classification.

Assume $v_1, \cdots, v_c$ are given in some way and suppose we wish to determine a classification rule of nearest prototype. The solution is evident:

$$\mathrm{U}_i(x) = \begin{cases} 1 & (v_i = \arg \min\limits_{1 \le j \le c} D(x, v_j)), \\ 0 & (\text{otherwise}). \end{cases} \tag{9}$$

For a technical reason we employ a closed ball $B(r)$ with the radius r:

$$B(r) = \{\ x \in \boldsymbol{R}^p\ :\ \|x\| \le r\ \}. \tag{10}$$

where r is sufficiently large so that it contains all prototypes $v_1, \cdots, v_c$, and we consider the problem inside this region.

Note that this function is the optimal solution of the following problem:

$$\min_{\mathrm{U}_j, j=1,\ldots,c} \sum_{j=1}^{c} \int_{B(r)} \mathrm{U}_j(x) D(x, v_j)\, dx \tag{11}$$

$$\text{subject to } \sum_{j=1}^{c} \mathrm{U}_j(x) = 1, \quad \mathrm{U}_\ell(x) \geq 0, \quad \ell = 1, \ldots, c. \tag{12}$$

We fuzzify this function. We note the above function is not differentiable. We 'regularize' the function by considering a differentiable approximation of this function.

For this purpose we add an entropy term and consider the following.

$$\min_{\mathrm{U}_j, j=1,\ldots,c} \sum_{j=1}^{c} \int_{B(r)} \mathrm{U}_j(x) D(x, v_j) dx + \nu \sum_{j=1}^{c} \int_{B(r)} \mathrm{U}_j(x) \log \mathrm{U}_j(x)/e dx \tag{13}$$

$$\text{subject to } \sum_{j=1}^{c} \mathrm{U}_j(x) = 1. \tag{14}$$

To obtain the optimal solution we employ the calculus of variations. Let

$$J = \sum_{j=1}^{c} \int_{B(r)} \mathrm{U}_j(x) D(x, v_j) + \nu \sum_{j=1}^{c} \int_{B(r)} \mathrm{U}_j(x) \log \mathrm{U}_j(x)/e$$

and notice the constraint. Hence we should minimize the Lagrangean

$$L = J + \int_{B(r)} \lambda(x) \{ \sum_{j=1}^{c} \mathrm{U}_j(x) - 1 \} dx.$$

Put $\mathbf{U}(x) = (\mathrm{U}_1(x), \ldots, \mathrm{U}_c(x))$ for simplicity. Let

$$\epsilon \delta L + o(\epsilon^2) = L(\mathbf{U} + \epsilon \zeta_i, \lambda) - L(\mathbf{U}, \lambda).$$

where

$$[\mathbf{U} + \epsilon \zeta_i](x) = (\mathrm{U}_1(x), \ldots, \mathrm{U}_{i-1}(x), \mathrm{U}_i(x) + \epsilon \zeta_i(x), \mathrm{U}_{i+1}(x), \ldots, \mathrm{U}_c(x)).$$

We then have

$$\begin{aligned} \delta L = & \int_{B(r)} \zeta_i(x) D(x, v_i) dx + \nu \int_{B(r)} \zeta_i(x) \{1 + \log \mathrm{U}_i(x)\} dx \\ & + \int_{B(r)} \zeta_i(x) \lambda(x) dx. \end{aligned}$$

Put $\delta L = 0$ and note $\zeta_i(x)$ is arbitrary. We hence have

$$D(x, v_i) + \nu(1 + \log \mathrm{U}_i(x)]) + \lambda(x) = 0$$

from which

$$\mathrm{U}_i(x) = \exp(-1 - \lambda(x)/\nu)\exp(-D(x, v_i)/\nu)$$

holds. Summing up the above equation with respect to $j = 1, \ldots, c$:

$$1 = \sum_{j=1}^{c} \mathrm{U}_j(x) = \sum_{j=1}^{c} \exp(-1 - \lambda(x)/\nu)\exp(-D(x, v_j)/\nu).$$

We thus obtain the solution

$$\mathrm{U}_i(x) = \frac{\exp(-D(x, v_i)/\nu)}{\sum_{j=1}^{c} \exp(-D(x, v_j)/\nu)}, \tag{15}$$

which is the same as (8) with $\ell = 2$.

For the classification function (8) with $\ell = 1$, the same type of the calculus of variations should be applied. Thus the problem to be solved is

$$\min_{\mathrm{U}_j, j=1,\ldots,c} \sum_{j=1}^{c} \int_{B(r)} (\mathrm{U}_j(x))^m D(x, v_j) dx \tag{16}$$

$$\text{subject to } \sum_{j=1}^{c} \mathrm{U}_j(x) = 1. \tag{17}$$

The optimal solution is, as we expect,

$$\mathrm{U}_i(x) = \frac{1/D(x, v_i)^{\frac{1}{m-1}}}{\sum_{j=1}^{c} 1/D(x, v_j)^{\frac{1}{m-1}}}. \tag{18}$$

We omit the derivation, as it is similar to the former derivation.

Let us consider a functional for the possibilistic clustering.

$$\min_{\mathrm{U}_j, j=1,\ldots,c} \sum_{j=1}^{c} \int_{B(r)} \{(\mathrm{U}_j(x))^m D(x, v_j) dx + \eta_i (1 - \mathrm{U}_j(x))^m\} \tag{19}$$

Note that no constraint is imposed for the possibilistic clustering. Let

$$J_i = \int_{B(r)} \{(\mathrm{U}_j(x))^m D(x, v_j) dx + \eta_i (1 - \mathrm{U}_j(x))^m\}$$

and note that J_i are independently minimized of other J_ℓ. Hence we should calculate

$$\epsilon \delta J_i + o(\epsilon^2) = J_i(\mathbf{U} + \epsilon\zeta) - J_i(\mathbf{U}).$$

We have

$$\delta J_i = m \int_{B(r)} \{(\mathrm{U}_j(x))^{m-1} D(x, v_j) - \eta_i (1 - \mathrm{U}_j(x))^{m-1}\}\zeta(x)$$

Put $\delta J_i = 0$ and noting $\zeta(x)$ is arbitrary, we have

$$(\mathrm{U}_j(x))^{m-1} D(x, v_j) - \eta_i (1 - \mathrm{U}_j(x))^{m-1} = 0,$$

form which we have

$$\mathrm{U}_j(x) = \frac{1}{1 + (D(x, v_j)/\eta_i)^{\frac{1}{m-1}}}.$$

The formulation by the calculus of variations thus justifies the use of these functions in the fuzzy learning and fuzzy classification. For example, we can consider a learning scheme (cf.[14]) with a fuzzy classification function of the above functions. In a fuzzy K-nearest neighbor method [9,2], either of $F(x; v_i)^{(\ell)}$ $(\ell = 2, 3)$ can be used for fuzzy classification.

As the last remark in this section, we note that the radius r in $B(r)$ can be arbitrarily large, and hence the classification function is assumed to be given in the whole space $\boldsymbol{R}^p$. The remark is valid for all classification functions discussed in this paper.

4 Generalizations

Let us consider a generalized formulation by introducing another function $\xi(x)$ into the functional. Consider

$$\min_{\mathrm{U}_j, j=1,\ldots,c} \sum_{j=1}^{c} \int_{B(r)} \mathrm{U}_j(x) \xi_j(x) D(x, v_j) dx + \nu \sum_{j=1}^{c} \int_{B(r)} \mathrm{U}_j(x) \log \mathrm{U}_j(x)/e dx \quad (20)$$

$$\text{subject to } \sum_{j=1}^{c} \mathrm{U}_j(x) = 1, \quad \mathrm{U}_\ell(x) \geq 0, \quad \ell = 1, \ldots, c. \quad (21)$$

The solution is

$$\mathrm{U}_i(x) = \frac{\exp(-\xi_i(x) D(x, v_i)/\nu)}{\sum_{j=1}^{c} \exp(-\xi_j(x) D(x, v_j)/\nu)}. \quad (22)$$

While the previous section with $\xi_j(x) \equiv 1$ leads to the fuzzy classification function, the function $\xi_j(x)$ can be taken to be a 'distribution of objects.' In order to represent ordinary objects $\{x_1, \ldots, x_n\}$, we can take

$$\xi_j(x) = \sum_{k=1}^{n} \delta(x - x_k)$$

where $\delta(x - x_k)$ is the delta function as the opposite extreme case. Consequently the problem is reduced to the minimization of $J_2(U, V)$ with respect to U.

What occurs to $\mathrm{U}_i(x)$ for $x \neq x_k$ $(k = 1, 2, \ldots, n)$? Equation (22) shows $\mathrm{U}_i(x) = 1/c$ which means we have no information and hence the equal

membership to every cluster is attached. This solution is acceptable but on the other hand there is no information $\mathrm{U}_i(x)$ from $J_2(U,V)$. To solve this inconsistency we put

$$\xi_j(x) = \sum_{k=1}^{n} \delta(x - x_k) + \varepsilon,$$

and take the limit of $\varepsilon \to 0$. We then have $\mathrm{U}_i(x_k) = u_{ik}$ and $\mathrm{U}_i(x) = 1/c$ for $x \neq x_k$ $(k = 1, 2, \ldots, n)$ from the formulation (20), as expected.

The other choice of the distribution of objects is to take uncertainty of object locations into account [3,4]. For example we can assume

$$\theta(x) = \begin{cases} 1 & (\|x\| \leq \gamma), \\ 0 & (\text{otherwise}), \end{cases}$$

($\gamma > 0$ is a given parameter) and put

$$\xi_j(x) = \sum_{k=1}^{n} \theta(x - x_k) + \varepsilon.$$

Then equation (22) gives the desired solution as well.

Another generalization is to change $D(x, v_i) = \|x - v_i\|^2$. For example suppose D is a region in $\boldsymbol{R}^p$. We take

$$D(x, v_1) = \begin{cases} \alpha & (x \in D), \\ \beta & (x \notin D), \end{cases}$$
$$D(x, v_i) = \|x - v_i\|^2, \quad i = 2, \ldots, c.$$

with $\alpha, \beta > 0$ and $\xi_i(x) \equiv 1$. Let α be a constant while $\beta \to \infty$. We then have

$$\mathrm{U}_i(x) = \begin{cases} \dfrac{\exp(-\alpha/\nu)}{\sum_{j=2}^{c} \exp(-D(x, v_j)/\nu) + \exp(-\alpha/\nu)} & (i = 1), \\ \dfrac{\exp(-D(x, v_i)/\nu)}{\sum_{j=2}^{c} \exp(-D(x, v_j)/\nu) + \exp(-\alpha/\nu)} & (i = 2, \ldots, c), \end{cases} \tag{23}$$

for $x \in D$, while

$$\mathrm{U}_i(x) = \begin{cases} 0 & (i = 1), \\ \dfrac{\exp(-D(x, v_i)/\nu)}{\sum_{j=2}^{c} \exp(-D(x, v_j)/\nu)} & (i = 2, \ldots, c). \end{cases} \tag{24}$$

This formulation is useful when noise cluster should be taken into account. The region D is the set in which noise can be contained.

Other generalizations are also possible. For example v_i can be not only the cluster center but also other types of cluster prototypes such as those in fuzzy c-varieties [1], fuzzy c-regression models [7], and fuzzy c-shell prototypes [8]. The formulation is similar to the case of a noise cluster shown above and we omit the detail.

5 Properties of Classification Functions

Investigation of the classification functions leads us to a number of observations on their theoretical properties. It is convenient to see the inverse of the function $1/\mathrm{U}_i(x)$ instead of $\mathrm{U}_i(x)$. That is, we have

$$\frac{1}{\mathrm{U}_i^{(1)}(x)} = 1 + \sum_{j \neq i} \frac{D(x, v_i)}{D(x, v_j)}, \tag{25}$$

$$\frac{1}{\mathrm{U}_i^{(2)}(x)} = 1 + \sum_{j \neq i} \exp\left(\nu^{-1}(D(x, v_j) - D(x, v_i))\right). \tag{26}$$

where $\mathrm{U}_i^{(1)}(x)$ and $\mathrm{U}_i^{(2)}(x)$ are respectively the classification functions for the standard fuzzy c-means and the entropy-based method. Notice in particular that the function $\mathrm{U}_i^{(1)}(x)$ is different from those in standard literature [1]. That is, singularity at $x = v_i$ is eliminated in (25).

For convenience the hard classification rule is named $\mathrm{U}_i^0(x)$. Thus,

$$\mathrm{U}_i^0(x) = \begin{cases} 1 & (v_i = \arg\min\limits_{1 \leq j \leq c} D(x, v_j)), \\ 0 & (\text{otherwise}). \end{cases} \tag{27}$$

It has been well-known that $\mathrm{U}_i^0(x)$ gives the Voronoi set with the center v_i [10]. The Voronoi set with the center v_i is denoted by $V(v_i)$. Moreover we make the fuzzy classification functions crisp by the maximum membership rule.

$$\mathcal{C}(\mathrm{U}_i^{(\ell)}(x)) = \begin{cases} 1 & (\mathrm{U}_i^{(\ell)}(x) > \mathrm{U}_j^{(\ell)}(x),\ \forall j \neq i), \\ 0 & (\text{otherwise}), \end{cases} \tag{28}$$

where $\ell = 1, 2$ and $\mathcal{C}(\cdot)$ is a mapping of $[0,1]^{\boldsymbol{R}^p}$ into $\{0,1\}^{\boldsymbol{R}^p}$.

We have the following propositions by observing (25) and (26) whose proofs are not difficult and omitted here (see [13]).

Proposition 1. *Given cluster centers $v_1, \ldots, v_c$, the function $\mathrm{U}_i^0(x)$ and $\mathcal{C}(\mathrm{U}_i^{(\ell)}(x))$ gives the same Voronoi sets:*

$$\mathrm{U}_i^0(x) = \mathcal{C}(\mathrm{U}_i^{(\ell)}(x)), \quad \forall x \in \boldsymbol{R}^p, \qquad \ell = 1, 2.$$

Proposition 2. *The classification function for the standard fuzzy c-means $\mathrm{U}_i^{(1)}(x)$ takes its maximum value 1 at $x = v_i$, while as $\|x\| \to \infty$, the function approaches to $1/c$:*

$$\mathrm{U}_i^{(1)}(v_i) = \max_{x \in \boldsymbol{R}^p} \mathrm{U}_i^{(1)}(x) = 1$$

$$\lim_{\|x\| \to \infty} \mathrm{U}_i^{(1)}(x) = \frac{1}{c}.$$

Proposition 3. *The classification function for the entropy-based method* $\mathrm{U}_i^{(2)}(x)$ *does not necessarily take its maximum value at* $x = v_i$. *Assume the Voronoi region* $V(v_i)$ *is bounded. Then*

$$\lim_{\|x\|\to\infty} \mathrm{U}_i^{(2)}(x) = 0.$$

On the other hand, if the Voronoi region $V(v_i)$ *is unbounded, there is a direction such that* $\|x\| \to \infty$ *and* $x \in V(v_i)$. *We have*

$$\lim_{\|x\|\to\infty} \mathrm{U}_i^{(2)}(x) = 1,$$

provided that none of two vectors $v_i - v_j$, $v_i - v_\ell$ $(j, \ell \neq i)$ *are on a line.*

In this way, investigation of the fuzzy classification function reveals a number of theoretical properties underlying methods of fuzzy c-means.

6 Conclusion

We have investigated fuzzy classification functions associated with different objective functions of fuzzy c-means and possibilistic clustering. The method of calculus of variations has been employed to derive the classification functions directly, whereby the natures of different classification functions have been uncovered. Moreover generalizations of the formulation have been studied and theoretical properties of the classification functions are investigated. Without a classification function, it is difficult to see such theoretical properties of the fuzzy c-means clustering.

The present method is useful for both theory and applications. As shown above, we can predict the behavior of different solutions of fuzzy c-means and possibilistic clustering by examining their classification functions, whereby we can select the best method for a particular application. As applications, we can design different classification algorithms using the functions discussed here.

Various generalizations of the formulation should further be studied. We have thus much room for future studies.

Acknowledgment

This study has partly been supported by the Grant-in-Aid for Scientific Research, Japan Society for the Promotion of Science, No.16300065.

References

1. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum, New York (1981)
2. Bezdek, J.C., Keller, J., Krishnapuram, R., Pal, N.R.: Fuzzy Models and Algorithms for Pattern Recognition and Image Processing. Kluwer, Boston (1999)

3. Dave, R.N.: Characterization and detection of noise in clustering. Pattern Recognition Letters 12, 657–664 (1991)
4. Davé, R.N., Krishnapuram, R.: Robust clustering methods: a unified view. IEEE Trans. Fuzzy Syst. 5(2), 270–293 (1997)
5. Dunn, J.C.: A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. J. of Cybernetics 3, 32–57 (1974)
6. Gelfand, I.M., Fomin, S.V.: Calculus of Variations. In: Silverman, R.A. (tr.) and (ed.) Prentice-Hall, Englewood Cliffs, NJ (1963)
7. Hathaway, R.J., Bezdek, J.C.: Switching regression models and fuzzy clustering. IEEE Trans. on Fuzzy Systems 1(3), 195–204 (1993)
8. Höppner, F., Klawonn, F., Kruse, R., Runkler, T.: Fuzzy Cluster Analysis. Wiley, Chichester (1999)
9. Keller, J.M., Gray, M.R., Givens, Jr., J.A.: A fuzzy k - nearest neighbor algorithm. IEEE Trans. on Syst. Man and Cybern. 15, 580–585 (1985)
10. Kohonen, T.: Self-Organizing Maps, 2nd edn. Springer, Heidelberg (1997)
11. Krishnapuram, R., Keller, J.M.: A possibilistic approach to clustering. IEEE Trans. on Fuzzy Syst. 1(2), 98–110 (1993)
12. Miyamoto, S., Mukaidono, M.: Fuzzy c-means as a regularization and maximum entropy approach. In: Proc. of the 7th International Fuzzy Systems Association World Congress (IFSA'97), June 25-30, 1997, Prague, Czech, vol.II, pp. 86–92 (1997)
13. Miyamoto, S.: Introduction to Cluster Analysis: Theory and Applications of Fuzzy Clustering, Morikita-Shuppan, Tokyo, 1999 (in Japanese)
14. Pal, N.K., Bezdek, J.C., Tsao, E.C.-K.: Generalized clustering networks and Kohonen's self-organizing scheme. IEEE Trans. on Neural Networks 4(4), 549–557 (1993)

CWC: A Clustering-Based Feature Weighting Approach for Text Classification

Lin Zhu[1], Jihong Guan[2], and Shuigeng Zhou[1]

[1] Department of Computer Science and Engineering, Fudan University, 200433, China
{zhulin,sgzhou}@fudan.edu.cn
[2] Department of Computer Science and Technology, Tongji University, 201804, China
jhguan@mail.tongji.edu.cn

Abstract. Most existing text classification methods use the vector space model to represent documents, and the document vectors are evaluated by the *TF-IDF* method. However, *TF-IDF* weighting does not take into account the fact that the weight of a feature in a document is related not only to the document, but also to the class that document belongs to. In this paper, we present a Clustering-based feature Weighting approach for text Classification, or *CWC* for short. *CWC* takes each class in the training collection as a known cluster, and searches for feature weights iteratively to optimize the clustering objective function, so the best clustering result is achieved, and documents in different classes can be best distinguished by using the resulting feature weights. Performance of *CWC* is validated by conducting classification over two real text collections, and experimental results show that *CWC* outperforms the traditional *K*NN.

Keywords: Text classification, clustering, feature weighting, *K*NN.

1 Introduction

As the dramatic expansion of the Internet and the World Wide Web continues, the amount of on-line text information grows so drastically that automatic text classification becomes a key technique for organizing and processing text data. Text classification is a supervised learning process, defined as assigning category labels (pre-defined) to new documents based on the likelihood suggested by a training set of labeled documents. It has been extensively studied in machine learning and information retrieval research areas. A number of approaches for text classification have been proposed, which include decision trees [1], regression models [2], *K*NN (*K*-Nearest Neighbor) classification [3], [4], Bayesian probabilistic methods [5], inductive rule learning [6], neural networks [7], Support Vector Machines [8], and boosting method [9], and some others [10], [11], to name a few.

Among these text classification methods, *K*NN is the simplest strategy that searches the *k*-nearest training documents to the test document and use the classes assigned to those training documents to decide the class of the test document. It is easy to implement for it needs no training stage, which is a must for most other

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 204–215, 2007.

classification methods. Furthermore, experimental studies show that the *K*NN method offers promising performance in text classification [10].

Similar to many other existing classification methods, *K*NN also adopts the vector space model (VSM) [12] to represent documents. Each document is represented by a vector (W_1, W_2, W_3, …, W_n) in the feature space, where W_i stands for the weight of the i^{th} feature. The document vectors are evaluated mainly by the *TF-IDF* method and its variants [13]. However, *TF-IDF* feature weighting has obvious drawback: it does not take into account the fact that the weight of a feature in a document is related not only to the document, but also to the class that the document belongs to. In almost all real document collections, feature distribution in one class is often quite different from that in other classes. What's more, the same feature may have different importance or relevance in different classes. Hence, it is not sufficient to use only *TF-IDF* feature weighting, for it treats each feature equally for all classes. However, considering that each feature may have different importance in different classes, *TF-IDF* weighting can not differentiate the importance of a feature in different classes, which is unfavorable for classification performance.

In this paper, we present a new feature weighting scheme for text classification. We term the new approach ***C***lustering-based feature ***W***eighting approach for text ***C***lassification, *CWC* for short. The *CWC* approach is simple but effective in that it assigns different weights for the same feature in different documents and different classes. The idea of *CWC* is as follows: given a training text collection, taking each class in the training collection as a known cluster, then searching for feature weights iteratively to optimize the clustering objective function, so that the best clustering result is achieved, which means that documents in different classes are best distinguished using the resulting feature weights. The resulting weights are used further for text classification. Considering that the resulting weights can optimally partition the training documents, it is naturally to think that these weights should be most suitable for classification task. We evaluate the classification performance of the proposed method by conducting experiments over two real text collections, *Reuters-21578* and *20- Newsgroups*, and experimental results show that *CWC* can achieve better classification performance than the traditional *K*NN without using our weighting method.

The remainder of this paper is organized as follows. Section 2 introduces some related work. Section 3 presents our clustering-based feature weighting approach for text classification in detail. Section 4 describes the experimental results for evaluating the proposed approach and comparing *CWC* with the traditional *K*NN. Finally, Section 5 summarizes the paper and highlights future research work.

2 Related Work

Vector Space Model is the commonly used model for document representation. That is, a document corresponds to an *n*-dimensional document vector. All document vectors form the document space. Given a document vector, its dimensional components indicate the corresponding features' weights that indicate the importance of these features in that document. For a document *d* in a training corpus *D*, it can be

represented by VSM as $\vec{d} = (w_1, w_2, ..., w_n)$. Here, $\vec{d}$ indicates the vector of document d; w_i is the weight of the i^{th} feature. The *TF-IDF* method is often used to evaluate feature weight as follows:

$$w_i = \frac{tf_i \times \log(N/n_i)}{\sqrt{\sum_{i=1}^{n} (tf_i)^2 [\log(N/n_i)]^2}} \tag{1}$$

Above, N is the total number of documents in D, tf_i is the occurrence frequency of the i^{th} feature in document d, and n_i is the number of documents where the i^{th} feature appears. Given two documents d_i and d_j, the similarity coefficient between them can be measured by the inner product of their corresponding document vectors as follows:

$$Sim(d_i, d_j) = \vec{d}_i \bullet \vec{d}_j \tag{2}$$

Feature selection or weighting is a crucial subtask for text classification, which can influence the scalability, efficiency and accuracy of the trained classifiers. In feature selection, the goal is to eliminate irrelevant features. This is done by assigning binary relevance weights to the features (1 for relevant, and 0 for irrelevant). Up to date, various methods have been proposed for feature selection, such as document frequency, information gain, mutual information, chi-square, correlation coefficient, relevancy score, etc. [14], [15], [16], [17], [18]. Feature weighting can be seen as an extension of the selection process where the features are assigned continuous weights which can be regarded as degrees of relevance or importance. In text classification, the combination of feature selection and feature weight computation is actually feature weighting. However, this kind of feature selection has an inherent drawback. That is, regardless of the distribution difference of each feature in different classes of the training collection, it produces a set of weights for all classes. To tackle the problem, we propose the Clustering-based Feature Weighting approach, and use it for text classification. In fact, we combine feature selection and feature weighting into the same framework. First, we do the normal feature selection as in other classification methods. Second, based on the results of feature selection above, we do further feature weighting based on clustering optimization to get the importance value of each feature k in class C_i as v_{ik} ($0 \le v_{ik} \le 1$), and represent the final weight of this feature to document D_j as $tf\_idf_{kj} * v_{ik}$.

*K*NN text classification is a well-known lazy learning approach, which has been extensively studied in machine learning and pattern recognition, and applied to text classification. The idea of *K*NN classification is quite simple: given a test document, the system finds the K nearest neighbors among training documents in the training corpus, and uses the classes of the K nearest neighbors to weight class candidates. The similarity score of each nearest neighbor document to the test document is used as the weight of the classes of the neighbor document. If several of the K nearest neighbors share a class, then the per-neighbor weights of that class are added together, and the resulting weighted sum is used as the likelihood score of that class. By sorting the scores of candidate classes, a ranked list is obtained, and by setting a threshold on the scores, binary class assignments are achieved. The usage of threshold is a kind of soft classification strategy, which can make a document belong to several classes. For

simplicity, in this paper we consider only binary classification problem, i.e. the class space has flat structure and all classes in question are semantically disjointed, and each document in the training corpus belongs to only one class, and each test document can be classified into only one class that has the highest weighted sum.

Feature weighting has also been employed in cluster analysis [19, 20]. In [19], the SKWIC (***S***imultaneous ***K***ey***W***ord ***I***dentification and ***C***lustering), which is a kind of *K*-Means based clustering approach but has better clustering results. The main idea of *SKWIC* is based on "Not all terms are considered equally relevant in a single category of text documents". The method learns a different set of keyword weights for each cluster, with which each document cluster can be characterized by a possibly different set of keywords and described compactly in terms of not only the attribute values, but also their relevance. The *SKWIC* approach is based on *k*-Means algorithm, thus it inherits most of the advantages of *k*-Mean-type clustering algorithms, such as ease of computation and simplicity.

Unlike *SKWIC*, our goal is text classification that uses clustering method to weight the features. Our *CWC* approach employs *SKWIC* for clustering and feature weighting by considering each class in the training collection as a known cluster, and searching for feature weights iteratively to optimize the clustering objective function, so as to achieve the best clustering result, and then utilizing the resulting feature weights for classification. In fact, we must not use the SKWIC method; we can use other clustering methods to achieve our goal, e.g., the method in [20, 21]. The key point of the *CWC* method is applying clustering for feature weighting in text classification.

3 Clustering-Based Feature Weighting for Text Classification

Given a classification task, the training corpus χ contains N documents that form C distinct classes. Each document is represented by an n-dimensional document vector, and each document class corresponds to the centroid vector of the vectors of all documents in the class. We treat the C classes as C established clusters. Let $V_i = [v_{ik}]$ stand for the relevance weight of the k^{th} feature in cluster C_i, and x_{jk} is the weight of the k^{th} dimension in document x_j, and c_{ik} is the k^{th} component of the i^{th} cluster's center vector. Here, $1 \le i \le C$, $1 \le j \le N$, and $1 \le k \le n$. Our goal is to determine the values $[v_{ik}]$ ($1 \le i \le C$ and $1 \le k \le n$), such that the objective function of the C established clusters is optimized. For text classification, the cosine similarity measure is used for distance computation between two vectors. As in the *SKWIC* approach [19], we define the dissimilarity between document x_j and the center vector of the i^{th} cluster as follows:

$$\tilde{D}_{wc_{ij}} = \sum_{k=1}^{n} v_{ik} D^{k}_{wc_{ij}} \tag{3}$$

where

$$D^{k}_{wc_{ij}} = \frac{1}{n} - (x_{jk} \bullet c_{ik}) \cdot \tag{4}$$

Since text classification is a kind of supervised learning, and each training document belongs to a certain known class, so we can determine each cluster center directly from the training text set, *i.e.*, the averaged vector of all training vectors that belong to the cluster, which can be described as follows.

$$c_{ik} = \frac{\sum_{x_j \in \chi_i} x_{jk}}{|\chi_i|} . \tag{5}$$

Therefore, $D^k_{wc_{ij}}$ represents for the distance between the k^{th} dimension document x_j to the cluster C_i, and $\tilde{D}_{wc_{ij}}$ stands for the weighted aggregate sum of cosine-based distances of document x_j to cluster C_i along the individual dimensions, *i.e.* the dissimilarity between document x_j and the cluster *Ci*. Here, we omit the normalization of $D^k_{wc_{ij}}$, for we assume that x_{jk} and c_{ik} are normalized to unit length before the evaluation of formula (4).

The aim of clustering based weighting of *CWC* is to search for an optimal set of feature weights. Each cluster *Ci* has its own set of feature weights $V_i = [v_{i1}, \cdots, v_{in}]$, and all clusters are internally compact under such feature weights. The objective function is defined similarly to that of *SKWIC* as follows:

$$J(C, V; \chi) = \sum_{i=1}^{C} \sum_{x_j \in C_i} \sum_{k=1}^{n} v_{ik} D^k_{wc_{ij}} + \sum_{i=1}^{C} \delta_i \sum_{k=1}^{n} v_{ik}^2 , \tag{6}$$

in which

$$\begin{cases} v_{ik} \in [0,1] & \forall i,k; \\ \sum_{k=1}^{n} v_{ik} = 1 & \forall i. \end{cases} \tag{7}$$

Above, the objective function is composed of two parts. The first component is the sum of weighted distances of all documents to the cluster centers, which is minimized when only one feature in each cluster is completely relevant ($v_{ik} = 1$), and all other features are irrelevant ($v_{ik} = 0$). The second component is the sum of the squared feature weights, which achieves global minimum when all features are equally weighted. When the two components are combined and all δ_i ($1 \le i \le C$) are chosen appropriately, we can get minimized objective function with the optimized feature weights.

With respect to the constraints in (7), we can get optimized J with the Lagrange multiplier technique, and have

$$J(\Lambda, V) = \sum_{i=1}^{C} \sum_{x_j \in C_i} \sum_{k=1}^{n} v_{ik} D^k_{wc_{ij}} + \sum_{i=1}^{C} \delta_i \sum_{k=1}^{n} v_{ik}^2 - \sum_{i=1}^{C} \lambda_i \left(\sum_{k=1}^{n} v_{ik} - 1\right) . \tag{8}$$

where $\Lambda = [\lambda_1, \ldots, \lambda_c]^t$.

Considering that the rows of *V* are independent to each other, we can simplify the minimization of $J(\Lambda, V)$ as minimizing each independent component as follows:

$$J_i(\lambda_i, V_i) = \sum_{x_j \in C_i} \sum_{k=1}^{n} v_{ik} D^k_{wc_{ij}} + \delta_i \sum_{k=1}^{n} v_{ik}^2 - \lambda_i (\sum_{k=1}^{n} v_{ik} - 1), \; i=1, \ldots, C . \tag{9}$$

Evaluating the gradient of J_i over v_{ik} and λ_i, and setting the gradients to zero, we have the following group of equations:

$$\begin{cases} \dfrac{\partial J_i(\lambda_i, V_i)}{\partial \lambda_i} = (\sum_{k=1}^{n} v_{ik} - 1) = 0 \\ \dfrac{\partial J_i(\lambda_i, V_i)}{\partial v_{ik}} = \sum_{x_j \in C_i} D^k_{wc_{ij}} + 2\delta_i v_{ik} - \lambda_i = 0 \end{cases} \tag{10}$$

Solving the above group of equations (10) for v_{ik}, we obtain

$$v_{ik} = \frac{1}{n} + \frac{1}{2\delta_i} \sum_{x_j \in X_i} [\frac{\sum_{k=1}^{n} D^k_{wc_{ij}}}{n} - D^k_{wc_{ij}}] . \tag{11}$$

The first part of v_{ik}, *i.e.* 1/*n*, is the default weight value when all features are regarded to have uniform distribution in the classes, while the second part are the weights considering the discriminations for non-uniform feature distributions. The bias may be positive when the distance along one dimension is less than the averaged total distance over all dimensions. That is to say, the feature is more compact than other features in the cluster; therefore the feature should be emphasized and assigned a relatively larger weight v_{ik}. Moreover, the $D^k_{wc_{ij}}$ is possible to be negative, which means that the feature distribution along this dimension is further more compact and the importance of this dimension should be emphasized further with larger v_{ik}.

The determination of δ_i value is also important for the objective function. If δ_i is too small, then only one feature in cluster C_i will be relevant and assigned a weight of one and all other features will be assigned quite small weights or even zero weights; while if δ_i is chosen too large, then all features in cluster C_i will be assigned equal weights as 1/*n*, and all features are considered to have the same importance, which is not reasonable in practice. Consequently, we will compute the optimal δ_i iteratively as follows.

$$\delta_i^{(t)} = K_\delta \frac{\sum_{x_j \in \chi_i} \sum_{k=1}^{n} v_{ik}^{(t-1)} (D^{k^{(t-1)}}_{wc_{ij}})}{\sum_{k=1}^{n} (v_{ik}^{(t-1)})^2} . \tag{12}$$

In equation (12), the superscript (*t*-1) denotes the results in iteration (*t*-1), K_δ is a constant and can be adjusted during the iteration process. If v_{ik} is often out of range [0, 1], which indicates that δ_i is too small, then K_δ should be increased. And if v_{ik} is nearly on the verge of 1/*n*, which means δ_i is too large, then the value of K_δ needs to

be reduced. If it occurs a few times that v_{ik} is negative, then we can adjust the negative feature weights according to the following formula:

$$v_{ik} \leftarrow v_{ik} + \left| \min_{k=1}^{n} v_{ik} \right| \quad if \quad v_{ik} < 0 \qquad . \tag{13}$$

The computation process of clustering based feature weighting of *CWC* is summarized in Algorithm 1, which is illustrated in Figure 1.

By using the clustering algorithm above to get a set of feature weights V_i for each cluster, we can compute the similarities between documents in the weighted feature space. Then the formula for the similarity between a test document $\vec{x}$ and a training document $\vec{d}_j$ is adjusted as follows:

$$sim(\vec{x}, \vec{d}_j) = \sum_{k=1, class(\vec{d}_j)=i}^{n} (v_{ik} \cdot x_k \cdot d_{jk}) \ . \tag{14}$$

Next, *CWC* can classify the documents based on *K*NN approach or other vector-based methods. By sorting the scores of candidate classes, we get a ranked list for the test document. Finally, by setting a threshold on the scores, binary class assignments are achieved.

Algorithm 1. Clustering-based Feature Weighting of *CWC*

Evaluate the center vector of each cluster $\vec{c}_i$ according to equation (5);

Initialize the feature weights v_{ik} to 1/*n*;

REPEAT {

Compute $D^k_{wc_{ij}}$, for $1 \le i \le C$, $1 \le j \le N$ and $1 \le k \le n$; according to equation (4);

Update the feature weights v_{ik} by using equation (11);

Adjust the value of v_{ik} according to equation (13);

Compute $\tilde{D}_{wc_{ij}}$, for $1 \le i \le C$, $1 \le j \le N$; according to equation (3);

Update δ_i, for $1 \le i \le C$ by using (12);

} UNTIL (v_{ik} stabilize);

Fig. 1. Clustering-based feature weighting algorithm

4 Experimental Results

This section provides empirical evidence that *CWC* outperforms the traditional *K*NN without using feature weighting. Experimental results are achieved by conducting classification over two real text collections: *Reuters-21578* and *20-Newsgroups*, which are two commonly used data sets for text classification performance evaluation.

4.1 Experimental Setup

Automatic text classification performance is usually measured in terms of precision (*Pr*) and recall (*Re*), which reflect the two aspects of classification quality. However, neither precision nor recall makes real sense when used separately. In this paper, in addition to *Pr* and *Re*, we also use a combined value of *Pr* and *Re*, i.e., *F1*, to evaluate classification performance. *F1* is evaluated in the following formula [16]:

$$F1 = \frac{Pr \times Re \times 2}{Pr + Re}. \tag{15}$$

To evaluate the average precision, recall and *F1*, there exist the micro and the macro averaging methods. Macro averaging is based on classes, and micro averaging is based on documents. Here, we provide both micro and macro-averaged precision, recall and *F1*.

Our *CWC* approach is implemented based on the open source classification tool "rainbow" [22]. In our experiments, the *k*-fold cross-validation method is applied to train and test the classifier. We divide the testing data set into 10 subsets evenly, and get $\{T_1, T_2, \ldots, T_{10}\}$, each of which stands for a subset, and let the testing set $T_{test}=T_i$, and the training set $T_{train}=T-T_i$, where $i=1, 2, \ldots, 10$. The averaged results of these 10 experiments are taken as the final results. The preprocessing of text collections consists of: filtering stop-words, stemming, and feature selection. We sort all features in a decreasing order according to their information gain values, and select the top *n* features for further weighting. By default, we let n=2,000.

4.2 Experimental Results of *Mini_newsgroups*

Mini_newsgroup1 and *mini_newsgroup2* are two manually-compiled subsets of the *20-Newsgroups*. *Mini_newsgroup1* consists of four classes of quite different content, and *Mini_newsgroup2* is composed of four classes of nearly similar content. Here, we set the parameters for *K*NN as K=20 and K_δ=0.001. The computation of optimal feature weights converges at the 12^{th} iteration. We compare the performance of *CWC* to that of the traditional *K*NN method. Experimental results are shown in Table 1 and Table 2 respectively. From the results in the two tables, we can clearly see that *CWC* outperforms traditional *K*NN. Especially, for the text collections containing classes of quite similar content (e.g. *mini_newsgroup2*), our CWC approach improves the traditional *K*NN method considerably.

4.3 Experimental Results of 20-Newsgroups

We then conduct experiments on the entire *20-Newsgroups* text collection. *20-Newsgroups* contains 20 classes of different news. We test how the micro-averaged precision changes with *k*'s value. The results for *CWC* and traditional *K*NN are shown in Figure 2, where *k*'s value increases from 1 to 50. We can see that two methods all

get precision improved considerably as K increases from 1 to 10, while the precision keeps stable as K increases from 10 to 50. Nevertheless, within the test score of K's value, not matter what K is, the precision of *CWC* outperforms traditional *K*NN by about 2.5%. Figure 3 shows the changing trend of micro-averaged precision with the number of selected features (or features set size). Here, we let K=20. From Figure 3, we can see that within our test range of feature size, no matter how much the feature size is, the precision of *CWC* are larger than that of the traditional *K*NN, and with the increasing of feature size, the difference is enlarging. This demonstrate the performance advantage our *CWC* approach.

Table 1. Macro-averaged classification perform ance on *mini_newsgroup1*(*CWC* vs. *K*NN)

class	macro-precision%		macro-recall%		macro-*F1*	
	CWC	***K*NN**	***CWC***	***K*NN**	***CWC***	***K*NN**
comp.os.ms-windows.misc	**99.11**	96.97	**100.00**	99.00	**.9955**	.9797
rec.autos	**99.01**	97.84	**99.60**	99.00	**.9930**	.9842
sci.med	98.70	**98.88**	**98.70**	96.00	**.9870**	.9742
talk.religion.misc	99.00	**99.10**	**99.00**	98.50	**.9900**	.9880
Averaged	**98.96**	98.20	**99.33**	98.13	**.9914**	.9815

Table 2. Macro-averaged classification performance on *mini_newsgroup2* (*CWC* vs. *K*NN)

class	macro-precision%		macro-recall%		macro-*F1*	
	CWC	***K*NN**	***CWC***	***K*NN**	***CWC***	***K*NN**
comp.graphics	**98.21**	95.04	**91.90**	89.20	**.9495**	.9203
comp.os.ms-windows.misc	**92.46**	89.58	**98.60**	89.80	**.9543**	.8969
comp.sys.ibm.pc.hardware	**95.53**	86.07	**92.80**	88.20	**.9415**	.8712
comp.sys.mac.hardware	**95.15**	89.94	**97.50**	91.80	**.9631**	.9086
Averaged	**95.34**	90.16	**95.20**	89.75	**.9521**	.8993

4.4 Experimental Results of Reuters-21578

Finally, we carry out experiments on the *reuters-21578* text collection. Reuters-21578 contains 11,098 documents that can be classified into 71 classes, form which 7,780 documents are used for training, and 3,309 documents for testing. For the training set, the averaged number of documents per class is 109.70, and for the testing set, the value is 46.61. The precision results of traditional *K*NN and *CWC* are shown in Figure 4, where we take K_δ=0.05 and let K increases from 1 to 50. From Figure 4, we can see that the precision are improved quickly as K increases from 1 to 10, and reaches the highest point when K is around 10, while the precision decreases after K >10. However, within the test score of K's value, no matter what value K takes, *CWC* outperforms traditional *K*NN obviously. And the advantage of *CWC* over traditional *K*NN becomes more evident as K increases.

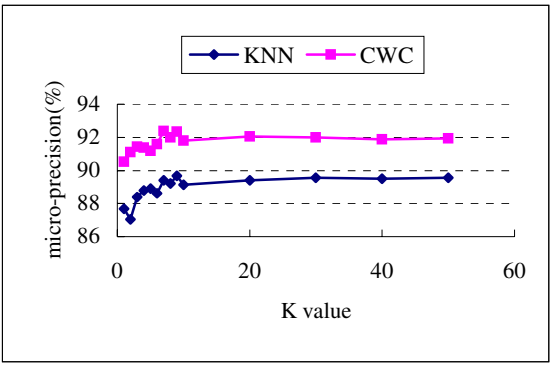


Fig. 2. Micro-precision *vs. K* (*20-Newsgroups*)

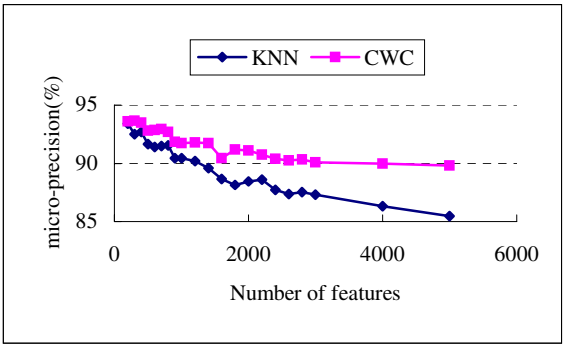


Fig. 3. Micro-precision *vs.* feature size (*20-Newsgroups*)

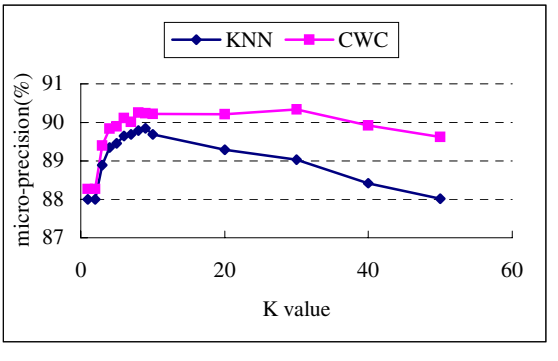


Fig. 4. Micro-precision *vs. K* (*Reuters-21578*)

5 Conclusion and Future Work

In this paper, we present the ***C***lustering-based feature ***W***eighting approach for text ***C***lassification approach, or *CWC* for short. *CWC* takes each class in the training collection as a known cluster, and searches for feature weights iteratively to optimize the clustering objective function, so the best clustering result is achieved. This also means that documents in different classes are best distinguished using the resulting feature weights. Performance of the proposed approach is validated by conducting experiments over two real text collections, and experimental results of classification performance show that *CWC* outperforms the traditional *K*NN evidently.

The *CWC* approach proposed in this paper adopts hard clustering model. However, in reality, the class of a document is usually quite fuzzy for it may belong to several different classes. That is, documents in a collection can rarely be described as members of a single or exclusive class. Most documents tend to straddle between two or more different classes, and assign such a document into a single class will affect the classification and retrieval abilities. Hence, to deal with real large document collections, further research is needed, such as fuzzy clustering methods for data set containing overlapping classes. Then each document will be labeled into several classes according to the relevance of the document to these classes. This is under consideration for our next steps. Further comparative studying of *CWC* with other weighting methods is another task of our future work.

Acknowledgements. This work was supported by National Natural Science Foundation of China (NSFC) under grants numbered 60573183 and 90612007. Jihong Guan was also supported by Program for New Century Excellent Talents in University of China under grant no. NCET-06-0376.

References

1. Apte, C., Weiss, S.: Data Mining with Decision Trees and Decision Rules. Future Generation Computer Systems 13, 197–210 (1997)
2. Yang, Y., Chute, C.G.: An Example-based Mapping Method for Text Categorization and Retrieval. ACM Transaction on Information Systems (TOIS) 12, 252–277 (1994)
3. Lam, W., Ho, C.Y.: Using a Generalized Instance Set for Automatic Text Categorization. In: Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 81–89 (1998)
4. Known, O.W., Lee, J.H.: Text categorization based on k-nearest neighbor approach for Web site classification. Information Processing and Management 39, 25–44 (2003)
5. Lewis, D.D.: Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval. In: Nédellec, C., Rouveirol, C. (eds.) Machine Learning: ECML-98. LNCS, vol. 1398, pp. 4–15. Springer, Heidelberg (1998)
6. Cohen, W.W., Singer, Y.: Context-Sensitive Learning Methods for Text Categorization. In: Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 307–315. ACM Press, New York (1996)
7. Wiener, E., Pedersen, J.O., Weigend, A.S.: A Neural Network Approach to Topic Spotting. In: Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval (SDAIR), pp. 317–332 (1995)

8. Joachims, T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In: Nédellec, C., Rouveirol, C. (eds.) Machine Learning: ECML-98. LNCS, vol. 1398, pp. 137–142. Springer, Heidelberg (1998)
9. Schapire, R.E., Singer, Y.: Improved Boosting Algorithms Using Confidence-rated Predictions. In: Proceedings of 11th Annual Conference on Computational Learning Theory, pp. 80–91 (1998)
10. Yang, Y., Liu, X.: A Re-examination of Text Categorization. In: Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 42–49. ACM Press, New York (1999)
11. Yang, Y.: An Evaluation of Statistical Approaches to Text Categorization. Journal of Information Retrieval 1, 67–88 (1999)
12. Salton, G., Wong, A., Yang, C.S.: A Vector Space Model Gets Automatic Indexing. In: Jones, K.S., Willett, P. (eds.) Readings in Information Retrieval, pp. 273–280. Morgan Kaufmann, San Francisco (1997)
13. Frakes, W.B., Baeza-Yates, R. (eds.): Information Retrieval: Data Structures-Algorithms. Prentice Hall PTR, Upper Saddle River, NJ, USA (1992)
14. Sebastiani, F.: Machine Learning in Automated Text Categorization. ACM Computing Surveys. 34, 1–47 (2002)
15. Yang, Y., Pedersen, J.: A Comparative Study on Feature Selection in Text Categorization. In: Proceedings of the 14th International Conference on Machine Learning, pp. 412–420 (1997)
16. Zheng, Z.H., Wu, X.Y., Srihari, R.: Feature Selection for Text Categorization on Imbalanced Data. SIGKDD Explorations. 6, 80–89 (2004)
17. Wang, G., Lochovsky, F.H., Yang, Q.: Feature Selection with Conditional Mutual Information MaxiMin in Text Categorization. In: Proceedings of CIKM'04. pp. 342–349, Washington, DC, USA (2004)
18. Forman, G.: An Extensive Empirical Study of Feature Selection Metrics for Text Classification. JMLR, Special Issue on Variable and Feature Selection. 3, 1289–1305 (2003)
19. Frigui, H., Nasraoui, O.: Simultaneous Clustering and Dynamic Keyword Weighting for Text Documents. In: Berry, M. (ed.) Survey of Text Mining, pp. 45–70. Springer, Heidelberg (2004)
20. Chan, E.Y., Ching, W.–K., Ng, M.K., Huang, J.: An optimization algorithm for clustering using weighted dissimilarity measures. Pattern Recognition 37(5), 943–952 (2004)
21. Torra, V., Miyamoto, S., Lanau, S.: Exploration of textual databases using a fuzzy hierarchical clustering algorithm in the GAMBAL system. Information Processing and Management 41(3), 587–598 (2005)
22. McCallum, A.K.: Bow: A Toolkit for Statistical Language Modeling, Text Retrieval, Classification and Clustering (1996), http://www.cs.cmu.edu/mccallum/bow

A Novel Spatial Clustering Algorithm with Sampling

Cai-Ping Hu, Xiao-Lin Qin, and Jun Zhang

College of Information Science and Technology,
Nanjing University of Aeronautics & Astronautics, Nanjing, 210016, China
hucaiping@nuaa.edu.cn

Abstract. Spatial clustering is one of the very important spatial data mining techniques. So far, a lot of spatial clustering algorithms have been proposed. *DBSCAN* is one of the effective spatial clustering algorithms, which can discover clusters of any arbitrary shape and handle the noise effectively. However, it has also several disadvantages. First, it does based on only spatial attributes, does not consider non-spatial attributes in spatial databases. Secondly, when *DBSCAN* does handle large-scale spatial databases, it requires large volume of memory support and the I/O cost. In this paper, a novel spatial clustering algorithm with sampling (*NSCAS*) based on *DBSCAN* is developed, which not only clusters large-scale spatial databases effectively, but also considers spatial attributes and non-spatial attributes. Experimental results of 2-D spatial datasets show that *NSCAS* is feasible and efficient.

1 Introduction

Spatial data mining refers to the extraction from spatial databases of implicit knowledge, spatial relations or significative features or patterns that are not explicitly stored in spatial databases[1]. Spatial data mining needs to integrate data mining and techniques of spatial database. It can be used for understanding spatial data, discovering spatial relationships and intrinsic relationships between spatial and non-spatial data, constructing spatial knowledge-bases, and reorganizing spatial databases, optimizing spatial queries, etc. It is expected to have wide applications in geographic information systems(GIS), remote sensing, image databases exploration, medical imaging, robot navigation, and other areas where spatial data are used.

Clustering is the process of grouping a set of objects into classes or clusters so that objects within a cluster have similarity in comparison to one another, but are dissimilar to objects in other clusters. So far, many clustering algorithms have been proposed, Based on a general definition, they can be categorized into five broad categories, i.e., hierarchical, partitional, density-based, grid-based and model-based[1]. (1) Partitional clustering methods[2]. For example, *CLARANS*. (2) Hierarchical clustering methods[3-4]. Such as *CURE* and *BIRCH*. (3) Density-based clustering methods[5-8]. Examples of density-based clustering methods include *DBSCAN*, *OPTICS, DBRS* and *GDBSCAN*. (4) Grid-based clustering methods[9-10]. Such as *STING* and *WaveCluster*. (5)Model-based clustering methods. For example, *COBWEB*. *DBSCAN* is one of

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 216–225, 2007.

the very effective spatial clustering algorithms, which can discover clusters of any arbitrary shape and handle the noise effectively. However, it has also several disadvantages. First, it does based on only spatial attributes, does not consider non-spatial attributes in spatial databases. Secondly, when *DBSCAN* does handle large-scale spatial databases, it requires large volume of memory support and the I/O cost. In this paper, a novel spatial clustering algorithm with sampling (*NSCAS*) based on *DBSCAN* is developed, which not only clusters large-scale spatial databases effectively, but also considers spatial attributes and non-spatial attributes.

The remainder of the paper is organized as follows: In section 2, we give an introduction to *DBSCAN* and analyze its disadvantages. Section 3 presents the *NSCAS* algorithm. Section 4 reports the experimental evaluation. Finally, Section 5 concludes the paper.

2 DBSCAN

DBSCAN is the first density-based spatial clustering method proposed [5].The key idea in *DBSCAN* is that to define a new cluster or to extend an existing cluster, a neighborhood around an object of a given radius (ε) must contain at least a minimum number of objects (*MinPts*), i.e., the minimum density for the neighborhood. The procedure for finding a cluster is based on the fact that a cluster is uniquely determined by any of its core objects.

To find a new cluster, *DBSCAN* starts from an arbitrary object q. It begins by performing a region query, which finds the neighborhood of the object q. If the neighborhood is sparsely populated, i.e. it contains fewer than *MinPts* objects, then the object q is labeled as noise. Otherwise, a cluster is created and all objects in q's neighborhood are placed in this cluster. Then the neighborhood of each of q's neighbors is examined to see whether it can be added to the cluster. If so, the process is repeated for every object in this neighborhood, and so on. If a cluster cannot be expanded further, *DBSCAN* selects another arbitrary unclassified object and repeats the process. This procedure is iterated until all objects in the dataset have been placed in clusters or labeled as noise. For a dataset containing *n* objects, n region queries are required. So its average time complexity is $O(n^2)$; If it uses an efficient spatial access data structure, such as an R*-tree[11], its the average time complexity is $O(nlogn)$.

In fact, for *DBSCAN* the process of clustering is an iterative procedure of executing region query, most of the time for clustering process is spending on region query operation. *DBSCAN* carries out region query operation for every object contained in the core object's neighborhood. However, executing all region queries to find these neighborhoods is very expensive. For a given core object p in cluster C, it's conceivable that the neighborhoods of the objects contained in p's neighborhood are most possibly intersecting with each other. Suppose q is an object in p's neighborhood, if its neighborhood is covered by the neighborhoods of other objects in p's neighborhood, then the neighborhood query operation for q can be omitted, which means that q is not necessary to be selected as a seed for cluster expansion. Therefore, the time consumed on region query operation for q can be cut down. As a matter of fact, for the dense

clusters, a lot of objects in a core object's neighborhood can be ignored being selected as seeds. So in order to speed up *DBSCAN* algorithm, we should take some representative objects rather than all objects in p's neighborhood as new seeds for the sake of reducing region queries.

Secondly, if non-spatial attributes play a important role in determining the desired clustering result, *DBSCAN* is not appropriate, because it does not consider non-spatial attributes. Spatial databases save a finite set of objects characterized by spatial and non-spatial attributes. The spatial attributes may represent, e.g., points or spatially extended objects such as polygons in the *d*-dimensional space .The non-spatial attributes of an object may represent additional properties of a spatial object, e.g., the unemployment rate for a community represented by a polygon in a geographic information system. In order to improve the quality of clustering, the clustering algorithm should not only depend on spatial attributes but also depend on non-spatial attributes.

3 A Novel Spatial Clustering Algorithm with Sampling

In this section, we present the *NSCAS* algorithm. In order to deal with non-spatial attributes, the concept of the matching neighborhood is proposed.

3.1 Basic Concepts

Given a dataset *DB*, a symmetric distance function *dist*, parameters ε and *MinPts*, and variable *attr* indicates the non-spatial attribute, the following definitions are used to specify *NSCAS*. (Extension to multiple non-spatial attributes is straightforward.)

Definition 1 (ε-neighborhood). The neighborhood of an object p, denoted by $N_\varepsilon(p)$, is defined as $N_\varepsilon(p) = \{q \in DB \mid dist(p,q) \le \varepsilon\}$.

Definition 2 (ε-matching neighborhood). The matching neighborhood of an object p, denoted by $N'_\varepsilon(p)$, is defined as $N'_\varepsilon(p) = \{q \in DB \mid dist(p,q) \le \varepsilon \text{ and } p.attr = q.attr\}$.

From the definition, it is obvious that the number of objects in the object p's ε-matching neighborhood must be fewer than or equal to that in its ε-neighborhood. Following the definition of *DBSCAN*, we give the definition of the *NSCAS* algorithm as follows.

Definition 3 (purity-core object). If the matching neighborhood of an object p has at least MinPts objects, i.e., $|N'_\varepsilon(p)| \ge MinPts$, the object p is a purity-core object

Definition 4 (directly purity-density-reachable). An object p is directly purity-density-reachable from an object q wrt.ε and MinPts, denoted by $p \Rightarrow q$, if

(1) $q \in N_{\varepsilon}^{'}(p)$

(2) $|N_{\varepsilon}^{'}(p)| \geq$ MinPts

Definition 5 (purity-density-connected). The purity-density-connected denoted by $\Leftrightarrow$ (wrt.ε and MinPts) is defined as the following:

(1) $\forall p \in DB,\ p \Leftrightarrow p$

(2) There is a chain of objects $p_1,p_2,,\ldots,p_n,p_1=q, p_n=p, p_i \in DB\ (i=1,2,\ldots,n)$, if $p_{i+1} \Rightarrow p_i$ or $p_i \Rightarrow p_{i+1}$,then $p \Leftrightarrow q$

From the definition, we can directly deduce the following important lemma.

Lemma 1. The relationship purity-density-connected $\Leftrightarrow$ wrt.ε and MinPts in dataset DB is an equivalent relationship.

For a non-empty dataset *DB*, because the relationship purity-density-connected $\Leftrightarrow$ is an equivalent relationship, it can be used to determine an unique partitioning of *DB*, each partitioning is a cluster or noise.

Definition 6. (Cluster) A cluster C wrt. ε and MinPts is a non-empty subset of dataset DB satisfying the following conditions:

(1) $\forall$ p, q $\in$ DB, if p $\in$ C and p $\Leftrightarrow$ q, then q $\in$ C;

(2) $\forall$ p, q $\in$ C, p $\Leftrightarrow$ q

3.2 Selection of Seeds

The selection of seeds is very important, the number of seeds should not be too small and too large. If it is too small, the neighborhoods of seeds can not completely cover the neighborhoods of other objects, objects lost and clusters divided may be yielded. Otherwise, unwanted region queries are executed, the advantage of the algorithm cannot be fully exploited. Here we will consider two-dimensional objects only. The method can be generalized for any possible dimensions greater than two. As shown in the Fig 1, with regard to ε and a purity-core object P, a circle is drawn with radius $\varepsilon$and center at object P. In addition, we draw the coordinate axes(xPy) pass through the center object P and the coordinate axes(x' P y') making the x' -axis an angle of 45^{o} to the x-axis. In the coordinate axes(xPy), the coordinates of the objects are (x,y),while the coordinates of them are (x' , y') in the coordinate axes(x' P y').

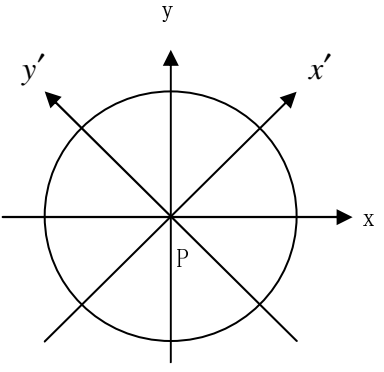


Fig. 1. Selection of seeds

The algorithm for selecting representative seeds from a purity-core object's neighborhood is proposed as the following, which is inspired by the *FDBSCAN* algorithm[12].The major difference between *FDBSCAN* algorithm and *NSCAS* algorithm is that in *NSCAS* algorithm the number of the representative seeds is eight while in the *FDBSCAN* algorithm it is four. Because the number of the representative seeds is only four, objects lost and clusters divided may be yielded. The key idea in the algorithm is that identifying the objects which their abscissa or ordinate is the biggest or the smallest in the corresponding coordinate axes and select them to be the representative seeds.

```
Finding_seeds(O,ε,Seeds)
Neighbours=DB.matchingNeighbours(O,ε);
Flag=i=1;
for each unclassified object p in the Neighbours do
if Flag==1then
   While i<=8 do
      seed_object[i]=p;
      i++;
      Flag++;
    endwhile
else
   if p.x>seed_object[1].x then seed_object[1]=p;
```

```
    if p.x' >seed_object[2].x' then seed_object[2]=p;
    if p.y>seed_object[3].y then seed_object[3]=p;
    if p.y' >seed_object[4].y' then seed_object[4]=p;
    if p.x<seed_object[5].x then seed_object[5]=p;
    if p.x' <seed_object[6].x' then seed_object[6]=p;
    if p.y<seed_object[7].y then seed_object[7]=p;
    if p.y' <seed_object[8].y' then seed_object[8]=p;
  endif
  Seeds=seed_object[1]∪seed_object[2]∪seed_object[3]∪
        seed_object[4]∪seed_object[5]∪seed_object[6]∪
        seed_object[7]∪seed_object[8];
  endfor
  end
```

3.3 Algorithm Description

Input: *DB*: database ; ε , *MinPts*: thresholds.

Output: Clusters of *DB*.

Algorithm NSCAS(*DB*, ε ,*MinPts*)

```
ClusterId=nextId(NOISE);
for each object O in DB do
   if O.ClId=UNCLASSIFIED
      if Expanding_cluster(O,ε,DB,MinPts,ClusterId)
           ClusterId=nextId(ClusterId);
end
```

```
Expanding_cluster(O,ε,DB,MinPts,ClId):Boolean
 //expanding clusters
Neighbours=DB.matchingNeighbours(O,ε);
if(|Neighbours|<MinPts)//O is a noise
   DB.changClId(O,NOISE);
 return false;
else//O is a purity-core object
   DB.changClId(Neighbours,ClId);
   Finding_Seeds(O,ε,Seeds);
   Seed_list=Seed_list.add(Seeds);
   While Seed_list≠NULL do
      currentP=Seed_list.first();
      Neighbours=DB.matchingNeighbours(currentP,ε);
     if(|Neighbours|≥MinPts)
     //currentP is a purity-core object
        Finding_Seeds(currentP,ε,Seeds);
        Seed_list=Seed_list.add(Seeds);
        for each object p in Neighbours do
          if p.ClId= UNCLASSIFIED or NOISE
               DB.changClId(p,ClId);
           else
                ClId=p.ClId;
// two purity-density-connected clusters are combined
         Seed_list.delete(currentP);
       Endif
     endwhile
```

```
      return true;
    endif
  end
```

The most important function used by *NSCAS* is *Expand_Cluster*, *Expand_Cluster* starts with an arbitrary unclassified object O and finds its matching neighborhood *Neighbours*. If the size of *Neighbours* is less than *MinPts*, then O is a noise. The algorithm finds its representative seed objects by the function of *Finding_seeds* for cluster expansion and adds them to *Seed_list*. After examining the matching neighborhood of O, the algorithm selects a representative seed object and repeats the above procedure for cluster expansion and then deletes the seed from *Seed_list*. The procedure is iterated until *Seed_list* becomes empty. If the cluster cannot be expanded further, the algorithm selects another arbitrary unclassified object and repeats the above procedure. The procedure is iterated until every object in the dataset is clustered or is labeled as noise.

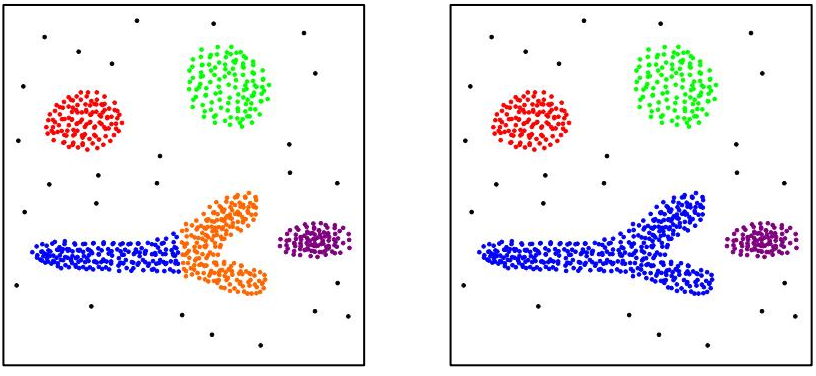
Compared with *DBSCAN*, *NSCAS* has mainly following improvement. Firstly, *NSCAS* can deal with non-spatial attributes. Because spatial databases save a finite set of objects characterized by spatial and non-spatial attributes, the clustering algorithm should not only depend on spatial attributes but also depend on non-spatial attributes. Secondly, in the function of *Expanding_cluster*, two purity-density-connected clusters can be combined by the sentence of *ClId=p.ClId*. Lastly, by adding the function of *Finding_seeds* into the function of *Expanding_cluster*, the algorithm needn't execute region query for all objects in a purity-core object's neighborhood, saving a lot of clustering time.

4 Experimental Evaluation

Here we evaluate the performance of *NSCAS* and compare it with the performance of *DBSCAN*, We have used both the synthetic database and the dataset, which was also used in Ref.[5]. All experiments were run on a 2.2 GHz PC with 256M memory.

4.1 Quality of the Clusters

To test the quality of the clusters, we applied two algorithms to a dataset used in Ref.[5]. In the dataset, the non-spatial attribute for the objects in the right part of the "Y" shaped dataset is different from that for those in left part. Fig 2(a) shows the clusters found by *NSCAS*, the "Y" shape at the right-bottom corner is divided into two clusters because of a difference in the values of the non-spatial attribute. The accuracy of *NSCAS* for this dataset is 100%. As shown in Figure 2(b),*DBSCAN* does not separate the cluster because it ignores the non-spatial attribute, the accuracy of *DBSCAN* for this dataset is lower than that of *NSCAS*.



(a) *NSCAS* (b) *DBSCAN*

Fig. 2. Clusters found by two algorithms

4.2 Efficiency

For comparison computational efficiency of *NSCAS* and *DBSCAN*, we used synthetic datasets that are consisted of objects from 20000 to 100,000. As shown in Figure 3, the performance of *NSCAS* is obviously better than that of *DBSCAN*. Using a spatial index such as an R*-tree, *NSCAS* has the average time complexity *O(nlogn)* as that of *DBSCAN*, where *n* is the size of the dataset. By sampling, *NSCAS* executes less region queries than *DBSCAN* does, which saves a lot of clustering time. Although *NSCAS* uses more one function than *DBSCAN* named *Finding_seeds*, Use of the function does not increase the overall complexity, the function of *Finding_seeds* used here has complexity *O(md)*, where *m* is the neighborhood size and *d* is the dimension of the object. The neighborhood size and dimension of an object are very small compared to the size of the database.

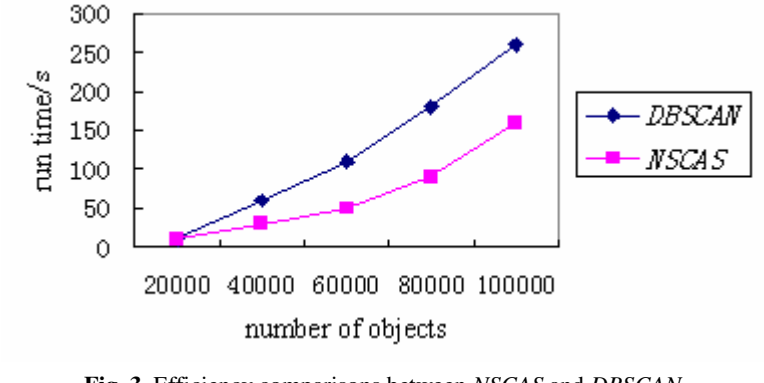


Fig. 3. Efficiency comparisons between *NSCAS* and *DBSCAN*

5 Conclusions

Based on *DBSCAN*, this paper presents a novel spatial clustering algorithm with sampling(*NSCAS*), which can also discover clusters of any arbitrary shape and handle the noise effectively. By sampling, *NSCAS* executes less region queries than *DBSCAN* does, which saves a lot of clustering time; As well, *NSCAS* can deal with non-spatial attributes, which improves clustering quality. Along with the amount of spatial data has been growing tremendously, we need further research how to improve the quality and the efficiency of clustering.

Acknowledgements. This research supported by the National Nature Science Foundation of China (No. 60673127) and the National Nature Science Foundation of Jiangsu Province (BK2001045).

References

1. Han, J., Kamber, M.: Data Mining Concepts and Techniques, Beijing, China. Machine Press (2001)
2. Ng, R.T., Han, J.: CLARANS: A Method for Clustering Objects for Spatial Data Mining. IEEE Transactions on Knowledge and Data Engineering 14(5), 1003–1016 (2002)
3. Guha, S., Rastogi, R., Shim, K.: CURE: An Efficient Clustering Algorithm For Large Databases. SIGMOD Record 27(2), 73–84 (1998)
4. Zhang, T., Ramakrishna, R., Livny, M.: BIRCH: An Efficient Data Clustering Method For Very Large Databases. SIGMOD Record 25(2), 103–114 (1996)
5. Ester, M., Kriegel, H., Sander, J., Xu, X.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: Proc. of 2nd KDD, Portland, pp. 226–231 (1996)
6. Ankerst, M., Breunig, M., Kriegel, H., Sander, J.: OPTICS: Ordering Objects to Identify the Clustering Structure. In: Proc. 1999 ACM SIGMOD Int. Conf. Management of Data Mining, PA, pp. 49–60. ACM Press, New York (1999)
7. Sander, J., Ester, M., Kriegel, H., Xu, X.: Density-based clustering in spatial databases: the algorithm GDBSCAN and its applications. In: Data Mining and Knowledge Discovery, vol. 2(2), pp. 169–194
8. Wang, X., Hamilton, H.J.: DBRS: A Density-Based Spatial Clustering Method with Random Sampling. In: Whang, K.-Y., Jeon, J., Shim, K., Srivastava, J. (eds.) PAKDD 2003. LNCS (LNAI), vol. 2637, pp. 563–575. Springer, Heidelberg (2003)
9. Zhang, W., Yang, Y., Munta, R.: STING: An Statistical Information Grid Approach to Spatial Data Mining. In: Proc. of 23rd VLDB Conf. Seattle, WA, pp. 186–195
10. Sheikholeslami, G., Chatterjee, S., Zhang, A.: WaveCluster: A multi resolution clustering approach for very large spatial databases. In: Proceedings of the 24th Conference on VLDB, New York, NY, pp. 428–439.
11. Beckmann, N., Kriegel, H.-P., Schneider, R., Seeger, B.: The R*-Tree: An Efficient and Robust Access Method for Objects and Rectangles[J]. SIGMOD Record 19(2), 322–331 (1990)
12. Zhou, S., Zhou, A., Jin, W., Fan, Y., Qian, W.: FDBSCAN: A Fast DBSCAN Algorithm. Journal of Software 11(6), 735–744 (2000)

Algorithms for Sequential Extraction of Clusters by Possibilistic Clustering

Sadaaki Miyamoto[1] and Youhei Kuroda[2]

[1] Department of Risk Engineering
Faculty of Systems and Information Engineering
University of Tsukuba, Ibaraki 305-8573, Japan
miyamoto@risk.tsukuba.ac.jp
[2] Graduate School of Systems and Information Engineering
University of Tsukuba, Ibaraki 305-8573, Japan
kuroda@soft.risk.tsukuba.ac.jp

Abstract. Possibilistic clustering that is robust to noise in data is another useful tool in addition to the best-known fuzzy c-means. However, there is a fundamental problem of strong dependence on initial values in possibilistic clustering and there is a proposal of an algorithm generating 'one cluster at a time.' Moreover this method is related to the mountain clustering algorithm. In this paper these features are reconsidered and a number of algorithms of sequential generation of clusters which includes a possibilistic medoid clustering are proposed. These algorithms automatically determine the number of clusters. An illustrative example with different methods of sequential clustering is given.

1 Introduction

As applications of data clustering are becoming more and more popular in a variety of research areas in sciences and engineering, many fuzzy clustering algorithms have been developed [1,2,5]. Among different methods of fuzzy clustering, the possibilistic clustering [8] is a useful tool from the viewpoint of robustness to noise in data [3] in addition to the best-known fuzzy c-means [4,1]. However, there are a fundamental problem of strong dependence on initial values in possibilistic clustering. Although there is a proposal of an algorithm generating 'one cluster at a time' [3] to mitigate this drawback, but an algorithm for such sequential generation of clusters is not yet fully studied.

In this paper problems in and around this method are investigated, whereby a family of new objective functions are derived which has the form of $J(V)$ of the variable of cluster centers instead of $J(U, V)$ of the two variables of memberships and the centers in the fuzzy c-means and possibilistic clustering in the original formulation.

Consequnetly, new algorithms of sequential generation of clusters are proposed from observation of the new objective functions. It will also be shown that these algorithms are related to the mountain clustering algorithm [13]. Moreover a medoid algorithm [6] related to possibilistic clustering is mentioned. An illustrative example is given to show properties of the proposed algorithms.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 226–236, 2007.

2 Objective Functions for Possibilistic Clustering

An object for clustering are denoted by $x_k = (x^1, \ldots, x_k^p) \in \boldsymbol{R}^p$, $(k = 1, \ldots, n)$, which is a vector in the p-dimensional Euclidean space. The set of object is denoted by $X = \{x_1, \ldots, x_n\}$. Cluster centers are $v_i = (v_i^1, \ldots, v_i^p)^T$, $i = 1, \ldots, c$, where c is the number of clusters. An abbreviated symbol $V = (v_1, \ldots, v_c)$ is used for the whole collection of cluster centers, while the matrix $U = (u_{ki})$, $(i = 1, \ldots, c,\ k = 1, \ldots, n)$ is used as usual, where u_{ki} means the degree of belongingness of object x_k to cluster i. In fuzzy c-means clustering, the constraint for a fuzzy partition is

$$M = \{U = (u_{ki}) : \sum_{i=1}^{c} u_{ki} = 1, \forall k; u_{kj} > 0, \forall j, k\},$$

while in possibilistic clustering we set

$$M = \{U = (u_{ki}) : u_{kj} \geq 0, \forall j, k\}. \tag{1}$$

The dissimilarity for clustering is the standard squared Euclidean distance between an individual and a cluster center:

$$D_{ki} = \|x_k - v_i\|^2.$$

We moreover use the dissimilarity between generic elements $x, y \in \boldsymbol{R}^p$, which is denoted by

$$D(x, y) = \|x - y\|^2.$$

Note 1. The constraint (1) is different from that in [8], but we observe the present constraint is easier to handle and has no harmful effect on the solutions.

2.1 Fuzzy c-Means and Possibilistic Clustering

As is well-known, fuzzy c-means [1] and possibilistic clustering are based on the optimization of an objective function. We consider two different types of objective functions.

$$J_e(U, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} \{u_{ki} D_{ki} + \lambda^{-1} u_{ki} (\log u_{ki} - 1)\} \tag{2}$$

$$J_2(U, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} \{(u_{ki})^2 D_{ki} + \zeta^{-1} (1 - u_{ki})^2\} \tag{3}$$

J_e is the entropy-based objective function [9], while J_2 is the function for the possibilistic clustering [8] which is restricted to $m = 2$ and $\eta_i = \zeta^{-1}$ $(1 \leq i \leq c)$.

Since we discuss the possibilistic clustering, the objective function of fuzzy c-means by Bezdek [1] and Dunn [4] is useless, since it gives the trivial solution of $u_{ki} = 0$ in possibilistic clustering.

In the next description of the alternative optimization **PCM** of fuzzy c-means clustering, we can use $J = J_e$ or J_2 and the constraint (1) for possibilistic clustering.

Algorithm PCM
PCM0. Set an initial value $\bar{V}$.
PCM1. Find optimal solution of J with respect to U while V is fixed: put

$$\bar{U} = \arg\min_{U \in M} J(U, \bar{V}).$$

PCM2. Find optimal solution of J with respect to V while U is fixed: put

$$\bar{V} = \arg\min_{V} J(\bar{U}, V).$$

PCM3. If the solution $(\bar{U}, \bar{V})$ is convergent, stop; else go to **PCM1**.
End of PCM

We show solutions of each step and we write, for simplicity, u_{ki} instead of $\bar{u}_{ki}$, v_i instead of $\bar{v}_i$, without confusion.

$$u_{ki} = \exp(-\lambda D_{ki}) \tag{4}$$

for J_e, or

$$u_{ki} = \frac{1}{1 + \zeta D_{ki}} \tag{5}$$

for J_2, while

$$v_i = \frac{\sum_{k=1}^{n} (u_{ki})^m x_k}{\sum_{k=1}^{n} (u_{ki})^m}. \tag{6}$$

in **PCM2** where $m = 1$ for J_e and $m = 2$ for J_2.

3 Sequential Extraction of Clusters

Let use define two functions related to (4) and (5).

$$U_e(x_k, y) = \exp(-\lambda D(x_k, y)) \tag{7}$$

$$U_2(x_k, y) = \frac{1}{1 + \zeta D(x_k, y)} \tag{8}$$

and note $U_e(x_k, v_i) = u_{ki}$ for J_e; $U_2(x_k, v_i) = u_{ki}$ for J_2.

In order to investigate properties of possibilistic clustering, we substitute $U(V) = (U_e(x_k, v_i))_{i=1,\ldots,c}$ into $J_e(U, V)$ where v_i is regarded as a variable. We have

$$J_e(U(V), V) = -\lambda \sum_{i=1}^{c} \sum_{k=1}^{n} \exp(-\lambda D(x_k, v_i)).$$

If we put $J'_e(V) = J_e(U(V), V)$ and

$$j_e(y) = -\sum_{k=1}^{n} \exp(-\lambda D(x_k, y)),$$

we have

$$J'_e(V) = J_e(U(V), V) = \lambda \sum_{i=1}^{c} j_e(v_i).$$

Although optimization of $J'_e(V)$ is not identical with the alternative minimization of $J_e(U, V)$, they are expected to have similar properties (cf. [12]). We hence investigate properties of $J'_e(V)$ as a function of V.

We immediately notice that $J'_e(V)$ is the sum of $j_e(v_i)$ and since there is no constraint on v_i, every $j_e(v_i)$ can be minimized independently from other $j_e(v_j)$ $(j \neq i)$. If we assume the minimizing element is unique, then the minimization leads to $\bar{v} = v_1 = \cdots = v_c$ and therefore only one cluster center will be obtained as the minimizing element of $J'_e(V)$. This means that if we want to have multiple clusters from this function, we should search several solutions of a multi-modal function at a time which is far more difficult than the minimization of a unimodal function.

This observation leads us to the idea of extracting 'one cluster at a time' which has been discussed by Davé and Krishnapuram [3]. We also note this idea has been given by them but not yet fully developed. Hence we discuss their idea in more detail, whereby we develop new algorithms.

We proceed to consider $J_2(U, V)$. Let us substitute $U(V) = (U_e(x_k, v_i))_{i=1,\ldots,c}$ into $J_2(U, V)$ in which v_i is a variable. We have

$$J'_2(V) = J_2(U(V), V) = \sum_{i=1}^{c} \sum_{k=1}^{n} \frac{D(x_k, v_i)}{1 + \zeta D(x_k, v_i)}.$$

We put

$$j_2(y) = \sum_{k=1}^{n} \frac{D(x_k, y)}{1 + \zeta D(x_k, y)},$$

and it follows that

$$J'_2(V) = \sum_{i=1}^{c} j_2(v_i).$$

We note again that $j_2(v_i)$ can be minimized independently from other $j_2(v_j)$. We thus have a unique solution $\hat{v} = v_1 = \cdots = v_c$, that minimizes $J'_2(V)$.

These observations justify the use of an algorithm to extract 'one cluster at a time.' A general and informal procedure for this is as follows.

1. Let the initial set of objects be $X^{(0)} = X$ and $k = 0$ Let the function $J(v; k) = j_e(v)$ (or $J(v; k) = j_2(v)$) with the set of objects $X^{(k)}$.
2. Search the minimizing element of $J(v; k)$:

$$v^{(k)} = \arg\min_{v} J(v; k)$$

3. Extract cluster $G^{(k)}$ that belongs to the center $v^{(k)}$.
4. Let $X^{(k+1)} = X^{(k)} - G^{(k)}$. If $X^{(k+1)}$ does not have sufficient elements to extract one more cluster, stop; else $k := k + 1$ and go to step 2.

A notable feature in this procedure is that we do not need to specify the number of clusters beforehand. However, the method of minimization is not specified in this procedure and hence we will consider this problem in the next section. Before considering the minimization schemes, we compare the above two objective functions and also review the mountain clustering. Note that a method of extracting one cluster at a time is called a sequential algorithm of possibilistic clustering here.

Note 2. The idea of 'one cluster at a time' has already been proposed in [3], but the above analysis, and algorithms shown below have not yet been studied.

3.1 Comparison of Objective Functions

In order to theoretically compare the objective functions $J_e'(V)$ and $J_2'(V)$, we investigate $j_e(y)$ and $j_2(y)$. Recall that $j_e(y)$ and $j_2(y)$ are respectively the sum of $-U_e(x_k, y)$ and $U_2(x_k, y)$:

$$j_e(y) = -\sum_{k=1}^{n} U_e(x_k, y),$$
$$j_2(y) = \sum_{k=1}^{n} U_2(x_k, y).$$

We observe that $U_e(x_k, y)$ and $U_2(x_k, y)$ as the function of y has similar properties in the minimization of $j_e(y)$ and $j_2(y)$.

Proposition 1. *The function $U_e(x_k, y)$ satisfies*

$$-U_e(x_k, x_k) = \min_y -U_e(x_k, y) = -1; \qquad \lim_{\|y\| \to \infty} -U_e(x_k, y) = 0,$$

while $U_2(x_k, y)$ satisfies

$$U_2(x_k, x_k) = \min_y U_2(x_k, y) = 0; \qquad \lim_{\|y\| \to \infty} U_2(x_k, y) = \frac{1}{\zeta}.$$

Moreover, if we put

$$g_e(z) = -\exp(-\lambda z), \quad g_2(z) = \frac{z}{1 + \zeta z}, \quad (0 \leq z < +\infty),$$

we see that the both functions $g_e(z)$ and $g_2(z)$ are monotonically increasing.

This proposition shows that we are handling objective functions of similar theoretical properties.

3.2 Mountain Clustering

Before stating minimization algorithms, we briefly review the mountain method [13], as this technique has a close relation to the present consideration. Indeed, the mountain clustering extracts sequentially, i.e., a cluster at a time.

This method considers the mountain function

$$M(y) = \sum_{k=1}^{n} \exp(-\alpha D(x_k, y)), \quad (\alpha > 0) \tag{9}$$

where $y \in \boldsymbol{R}^p$ is restricted to grid points. Let $y^{(1)}$ be the maximizing point of (9). Then the second mountain function is defined:

$$M^{(2)}(y) = M(y) - M(y^{(1)}) \sum_{k=1}^{n} \exp(-\alpha D(y^{(1)}, y)).$$

and then the calculation is repeated:

$$M^{(\ell)}(y) = M(\ell - 1)(y) - M(y^{(\ell-1)}) \sum_{k=1}^{n} \exp(-\alpha D(y^{(\ell-1)}, y)). \tag{10}$$

until there is no significant cluster. The stopping criterion is given by the ratio and a given parameter $\delta > 0$:

$$\frac{M(y^{(1)})}{M(y^{(\ell-1)})} < \delta. \tag{11}$$

We see that $M(y)$ is the same as $-j_e(y)$ and also the clusters are extracted one by one. Thus the idea of the mountain method is closely related to the present discussion.

4 Algorithms of Sequential Extraction of Clusters

The main problem in the above procedure is how to optimize the function $J(v; k)$, which we consider in this section.

Three procedures for the minimization are described in order to optimize $J(v; k)$.

First procedure is simplest. It minimizes the objective function on the finite set $\{y_1, \ldots, y_L\}$.

Procedure A

A1. Generate candidate points $y_1, \ldots, y_L \in \boldsymbol{R}^p$. $X^{(0)} = X$ and $k = 0$.

A2. Find minimizing element

$$\bar{y} = \arg \min_{v=y_1,\ldots,y_L} J(v; k).$$

A3. Find the cluster $G^{(k)}$ with the center $\bar{y}$. Extract $G^{(k)}$: $X^{(k+1)} = X^{(k)} - G^{(k)}$. If $X^{(k+1)}$ does not have sufficient elements to extract one more cluster, stop; else $k := k + 1$ and go to **A2**.

The points $y_1, \ldots, y_L$ can taken at grid points which is similar to the mountain method, or can be randomly chosen from X. Alternatively, we can take $\{y_1, \ldots, y_L\} = X$, then the method may be called a one-pass algorithm of *sequential possibilistic medoid* calculation, since the medoid is the cluster center that corresponds to an object (cf. Kaufman, Rousseeuw [6] for hard c-medoid).

The second procedure which is similar to the ordinary alternative minimization of fuzzy c-means is also useful, but it requires more calculation than the first method.

Procedure B

B1. Generate candidate points $y_1, \ldots, y_L \in \boldsymbol{R}^p$ and they are initial cluster centers. $X^{(0)} = X$ and $k = 0$.

B2. Repeat calculation of u_{ki} and v_i until convergence. Converged points are denoted by $z_1, \ldots z_\ell$. Find minimizing element

$$\bar{z} = \arg \min_{v=z_1,\ldots,z_\ell} J(v; k). \tag{12}$$

B3. Find the cluster $G^{(k)}$ with the center $\bar{z}$. Extract $G^{(k)}$: $X^{(k+1)} = X^{(k)} - G^{(k)}$. If $X^{(k+1)}$ does not have sufficient elements to extract one more cluster, stop; else $k := k + 1$ and go to **B2**.

It has been observed that several cluster centers are simultaneously obtained if we do not impose the condition (12), and hence this procedure is also useful as an algorithm of ordinary possibilistic clustering in general (cf. [11]).

4.1 Possibilistic Medoid Calculation

We can moreover mention the next procedure which may be called a multi-pass possibilistic medoid clustering in a generalized form.

Procedure C

C1. Generate candidate points $y_1, \ldots, y_L \in X$ and choose initial cluster centers $z_1, \ldots, z_c$ from $Y = \{y_1, \ldots, y_L\}$. $X^{(0)} = X$ and $k = 0$.

C2. Repeat **C3** until convergence.

C3. Let $y_{i1}, \ldots, y_{ki} \in Y$ be K-nearest elements to z_i $(i = 1, \ldots, c)$. Find minimizing element

$$\bar{z}_i = \arg \min_{v=z_i,y_{i1},\ldots,y_{ki}} J(v; k).$$

Put $z_i = \bar{z}_i$.

C4. Let

$$\bar{z} = \arg \min_{v=z_i, y_{i1}, \ldots, y_{ki}} J(v; k).$$

Find the cluster $G^{(k)}$ with the center $\bar{z}$. Extract $G^{(k)}$: $X^{(k+1)} = X^{(k)} - G^{(k)}$. If $X^{(k+1)}$ does not have sufficient elements to extract one more cluster, stop; else $k := k + 1$ and go to **C2**.

This procedure requires more calculation than the procedures A and B, and hence seems less useful for the sequential possibilistic clustering, but in order to find a medoid as a cluster center, such a procedure should be considered. With a slight modification, Procedure C is used for ordinary possibilistic medoid clustering. We thus have the next procedure.

Procedure C' (ordinary possibilistic medoid clustering)
C'1. Generate candidate points $y_1, \ldots, y_L \in X$ and choose initial cluster centers $z_1, \ldots, z_c$ from $Y = \{y_1, \ldots, y_L\}$.
C'2. Repeat **C'3** until convergence.

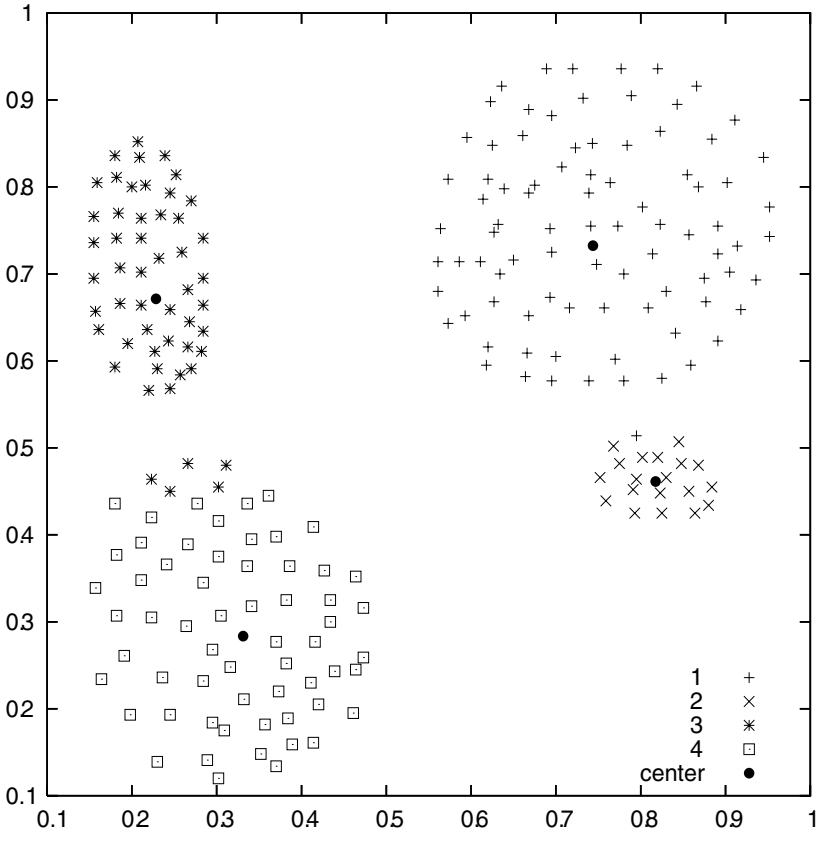


Fig. 1. Clusters $1 \sim 4$ sequentially extracted from a set of points on the plane using Procedure B

C'3. Let $y_{i1}, \ldots, y_{ki} \in Y$ be K-nearest elements to z_i $(i = 1, \ldots, c)$. Calculate

$$\bar{u}_{ki} = \exp(-\lambda D(x_k, z_i)), \quad \forall k, i,$$
$$\bar{V} = \arg \min_{v_i = z_i, y_{i1}, \ldots, y_{ki}} J(\bar{U}, V).$$

Put $z_i = \bar{v}_i$ $(i = 1, \ldots, c)$.

5 An Illustrative Example

Figures 1 and 2 are respectively the clusters obtained from Procedures B and C for the same set of points on the plane. The objective function J'_e with has been used and the extraction of objects uses the crisp criterion of

$$G^{(k)} = \{x_\ell \in X : \exp(-\lambda D(x_\ell, v^{(k)})) \geq \beta\}.$$

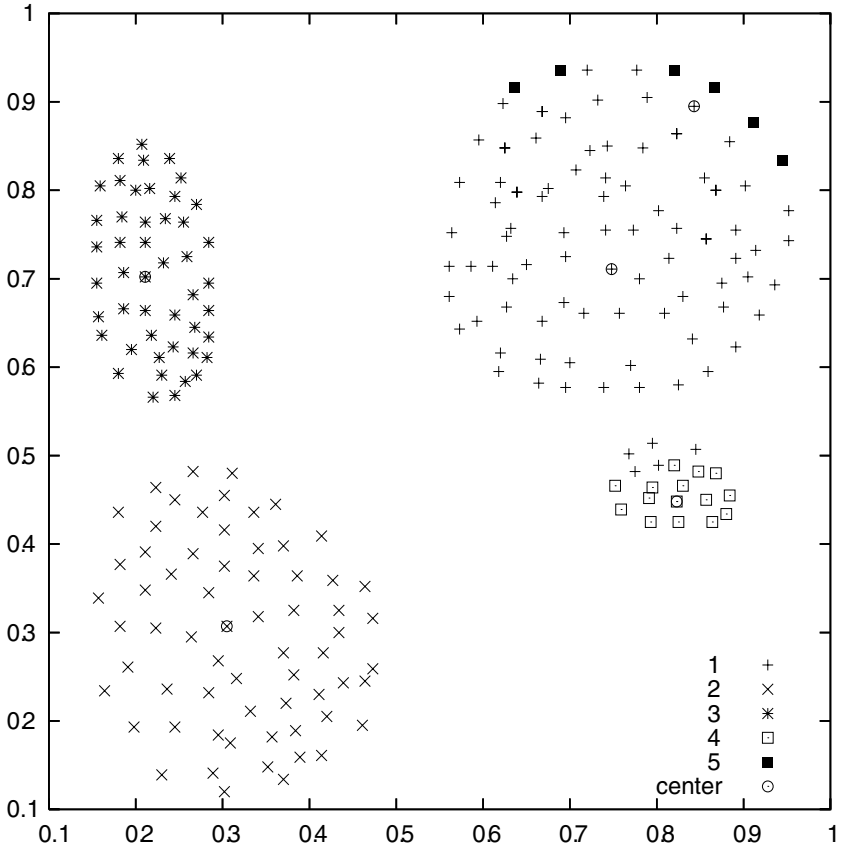


Fig. 2. Clusters $1 \sim 5$ sequentially extracted from a set of points on the plane using Procedure C

with the parameter $\beta = 0.2$.

The numbers $1, \ldots, 5$ at the lower right corner implies the number of the clusters sequentially extracted with the centers shown by small circles: the number 1 is the first cluster, the number 2 is the second, and so on.

In Figure 1, clusters $1 \sim 4$ have successfully extracted with the exception of several misclassified points shown by $*$ and $+$. After the fourth cluster havs been extracted, no object remains. We note that the correct number of the four clusters have been extracted although the number of clusters has not been given beforehand.

In Figure 2, the medoid algorithm has been used with all objects as $\{y_1, \ldots, y_L\}$ in Procdure C; the result is similar to that in Figure 1 except that the fifth cluster has been detected, but it is not well-separated from other clusters. Thus, this algorithm fails to find the correct number of clusters, but overall performance is acceptable.

Generally the medoid method does not perform as well as the method of c-means, as this figure shows, although medoids may be preferred in some real applications.

Note 3. We have omitted the result from Procedure A. It is similar to that from Procedure C. Generally Procedure C requires more calculation than Procedure A, but the results are more stable, since an iterative algorithm is used.

6 Conclusion

We have studied several algorithms of sequential extraction of clusters which connects the idea of possibilistic clustering [8,3] and the mountain clustering [13] by eliminating the membership matrix considering objective function of the cluster centers alone. We have also considered medoid algorithm using the sequential extraction and the possibilistic method.

The present study has uncovered the nature and properties of the method of possibilistic clustering and its variations. An implication of this study is that the mountain clustering has other options of taking random points instead of the grid points [13], and moreover $J_2'(V)$ can be used for the objective function in the mountain method. As another proposal we have mentioned the use of possibilistic medoids. Although we could not show an advantage of the medoid herein, applications such as document retrieval may prefer a medoid as a representative object of a cluster than a centroid.

In summary, possibilistic clustering as a sequential algorithm should be remarked, as the good property of the automatic determination of the number of clusters in this method should not be overlooked. Moreover, as there are many variations of fuzzy c-means, we have further investigations to be done in both methodological features and applications with regard to the present method.

Acknowledgment

This study has partly been supported by the Grant-in-Aid for Scientific Research, Japan Society for the Promotion of Science, No.16300065.

References

1. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum, New York (1981)
2. Bezdek, J.C., Keller, J., Krishnapuram, R., Pal, N.R.: Fuzzy Models and Algorithms for Pattern Recognition and Image Processing. Kluwer, Boston (1999)
3. Davé, R.N., Krishnapuram, R.: Robust clustering methods: a unified view. IEEE Trans. Fuzzy Syst. 5(2), 270–293 (1997)
4. Dunn, J.C.: A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. J. of Cybernetics 3, 32–57 (1974)
5. Höppner, F., Klawonn, F., Kruse, R., Runkler, T.: Fuzzy Cluster Analysis. Wiley, Chichester (1999)
6. Kaufman, L., Rousseeuw, P.J.: Finding Groups in Data: An Introduction to Cluster Analysis. Wiley, Chichester (1990)
7. Kohonen, T.: Self-Organizing Maps, 2nd edn. Springer, Heidelberg (1997)
8. Krishnapuram, R., Keller, J.M.: A possibilistic approach to clustering. IEEE Trans. on Fuzzy Syst. 1(2), 98–110 (1993)
9. Miyamoto, S., Mukaidono, M.: Fuzzy c-means as a regularization and maximum entropy approach. In: Proc. of the 7th International Fuzzy Systems Association World Congress (IFSA'97), Prague, Czech, June 25-30, 1997, vol. II, pp. 86–92 (1997)
10. Miyamoto, S.: Introduction to Cluster Analysis, Morikita-Shuppan, Tokyo, 1999 (in Japanese)
11. Miyamoto, S., Inokuchi, R., Kuroda, Y.: Possibilistic and Fuzzy c-Means Clustering with Weighted Objects. In: Proc. of 2006 IEEE International Conference on Fuzzy Systems, Vancouver, BC, Canada, July 16-21, 2006, pp. 4260–4265 (2006)
12. Runkler, T.A., Katz, C.: Fuzzy Clustering by Particle Swarm Optimization. In: IEEE International Conference on Fuzzy Systems, Vancouver, BC, Canada, July 16-21, 2006, pp. 3065–3072 (2006)
13. Yager, R.R., Filev, D.: Approximate clustering via the mountain method. IEEE Trans. on Syst. Man, and Cybern. 24(8), 1279–1284 (1994)

Fuzzy c-Means for Data with Tolerance Defined as Hyper-Rectangle

Yasushi Hasegawa[1], Yasunori Endo[2], Yukihiro Hamasuna[1], and Sadaaki Miyamoto[2]

[1] Graduate School of Systems and Information Engineering, University of Tsukuba, Japan
{hasegawa,yhama}@soft.risk.tsukuba.ac.jp
[2] Department of Risk Engineering, Faculty of Systems and Information Engineering, University of Tsukuba, Japan
{endo, miyamoto}@risk.tsukuba.ac.jp

Abstract. The paper presents some new clustering algorithms which are based on fuzzy c-means. The algorithms can treat data with tolerance defined as hyper-rectangle. First, the tolerance is introduced into optimization problems of clustering. This is generalization of calculation errors or missing values. Next, the problems are solved and some algorithms are constructed based on the results. Finally, usefulness of the proposed algorithms are verified through numerical examples.

1 Introduction

Clustering is one of the unsupervised classification and fuzzy c-means (FCM)[1] is one of the typical technique of fuzzy clustering.

In general, each data on a real space is transformed to a point in a pattern space and analyzed in clustering. However, the data should be often represented not by a point but by a set because of uncertainty of the data, e.g., measurement error margin, data that cannot be regarded as one point, and missing values in data. In the past, these uncertainties of data have been represented as interval range and many clustering algorithms for these interval ranges of data have been constructed[2,3] and one of the authors have also proposed one of such algorithms[4,5]. In these algorithms, nearest neighbor distance, furthest neighbor distance and Hausdorff distance have been used to calculate the dissimilarity between the target data in clustering. However, the guideline to select the available distance in each case has not been shown so that this problem is difficult. When we consider such a situation, it is more desirable to calculate the dissimilarity between such interval ranges of data without introducing a particular distance, e.g., nearest neighbor one and so on.

One of the authors has introduced the new concept of tolerance, which includes the above-mentioned uncertainties of data. The tolerance is different from the interval from the viewpoint of introduction of tolerance vectors. However he proposed two clustering algorithms, one is based on Euclidean norm[6] and the other is L_1-norm[7]. The tolerance is defined as hyper-sphere in these algorithms.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 237–248, 2007.

In this paper, we consider new optimization problems in which the tolerance is defined as hyper-rectangle and we construct new clustering algorithms based on sFCM (Standard Fuzzy c-means)[1] and eFCM (Entropy regularized Fuzzy c-means)[8] on L_1-norm and Euclidean (L_2) norm for data with tolerance through solving the optimization problems. It is easier to represent the uncertainty by hyper-rectangle than hyper-sphere because each objective attribute and the axis of pattern space are in one-to-one correspondence.

2 Theory

In this section, we discuss about optimization problems for clustering.

We define some notations at the beginning. $X = \{x_1, \ldots, x_n\}$ is a subset on a p dimensional vector space $\mathbf{R}^p$ and we write $x_k = (x_{k1}, \ldots, x_{kp})^T \in \mathbf{R}^p$. Here, we consider classifying the data set X into clusters $C_i (i = 1, \ldots, c)$. Let $v_i = (v_{i1}, \ldots, v_{ip})^T \in \mathbf{R}^p$ be the cluster center C_i and $V = \{v_1, \ldots, v_c\}$ be the set of cluster centers. Moreover, μ_{ki} is the membership grade belonging x_k to C_i and we denote the partition matrix $U = [\mu_{ki}]$. Fuzzy c-means calculates V and U which minimize an objective function by alternative optimization.

Here, we define tolerance vector $\varepsilon_k = (\varepsilon_{k1}, \ldots, \varepsilon_{kp})^T \in \mathbf{R}^p$, and $E = \{\varepsilon_1, \ldots, \varepsilon_n\}$ is a set of tolerance vectors. In the conventional works, the data has been represented as x_k. On the other hand, the data in this paper is allowed to be $x_k + \varepsilon_k$ because of the uncertainty. Simply speaking, the tolerance vector ε_k means divergence of the data from x_k. The constraint condition is shown by the following expression.

$$|\varepsilon_{kj}| \leq \kappa_{kj}, \ (\kappa_{kj} > 0). \tag{1}$$

Fig. 1 shows an example of tolerance vector in $\mathbf{R}^2$.

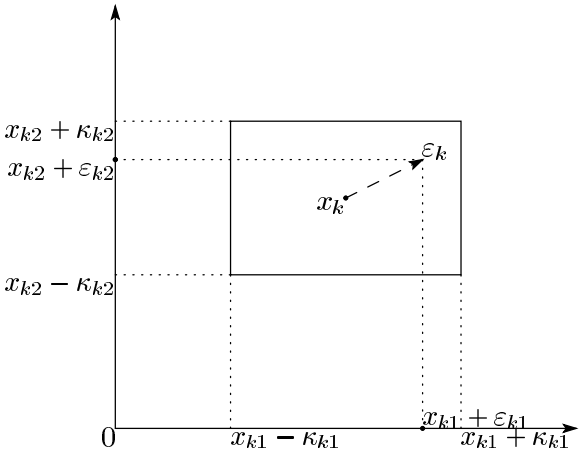


Fig. 1. An example of tolerance vector in $\mathbf{R}^2$

The purposes of the proposed algorithms are calculation of U, V and E by alternate optimization. In the next session, we will construct clustering algorithms for data with tolerance through mathematical discussion.

2.1 sFCM for Data with Tolerance Based on L_1-Norm

The objective function of sFCM based on L_1-norm is defined by Jajuga[9].

$$J_{\text{sFCM-}L_1} = \sum_{i=1}^{c}\sum_{k=1}^{n}\sum_{j=1}^{p}\mu_{ki}^{m}|x_{kj} - v_{ij}|,$$

under the constraint

$$\sum_{i=1}^{c}\mu_{ki} = 1,\ (\mu_{ki} \geq 0). \tag{2}$$

We define the following objective function based on the above equation.

$$J_{\text{sFCMT}_\text{R}\text{-}L_1} = \sum_{i=1}^{c}\sum_{k=1}^{n}\mu_{ki}^{m}d_{ki}, \tag{3}$$

where

$$d_{ki} = \sum_{j=1}^{p}|x_{kj} + \varepsilon_{kj} - v_{ij}|.$$

The following optimal solution is obtained by using the Lagrange function.

$$\mu_{ki} = \left(\sum_{s=1}^{c}\left(\frac{d_{ki}}{d_{ks}}\right)^{\frac{1}{m-1}}\right)^{-1}. \tag{4}$$

Here we propose two methods to obtain v_{ij}. One is based on Ref.[10] and called **Method 1**. The other is based on Ref.[9] and called **Method 2**.

Method 1

From (3), semi-objective function is

$$J_{ij}(v_{ij}) = \sum_{k=1}^{n}\mu_{ki}^{m}|x_{kj} + \varepsilon_{kj} - v_{ij}|.$$

If this equation is minimized, the objective function is also minimized. The optimal solution of v_{ij} is calculated according to the following procedures.

Step 1. Data is sorted in ascending order in each dimension.

$$x_{1j} + \varepsilon_{1j}, \ldots, x_{nj} + \varepsilon_{nj}$$

$$\downarrow \text{Sorting}$$

$$x_{q(1)j} + \varepsilon_{q(1)j} \leq \ldots \leq x_{q(n)j} + \varepsilon_{q(n)j}$$

where $q(k)$ is substitution of $(1, \ldots, n)$.

Step 2. We calculate as follows.

$$S = -\frac{1}{2}\sum_{k=1}^{n}(\mu_{ki})^m.$$

Step 3. It starts from $r = 0$ and the following calculations are repeated between $S < 0$.

$$r := r + 1;$$
$$S := S + (\mu_{q(r)i})^m;$$

Step 4. From the above calculation, we obtain

$$v_{ij} = x_{q(r)j} + \varepsilon_{q(r)j}. \tag{5}$$

Method 2

From (3), semi-objective function is

$$J_{ij}(v_{ij}) = \sum_{k=1}^{n} w_{ki}(x_{kj} + \varepsilon_{kj} - v_{ij})^2, \tag{6}$$

where

$$w_{ki} = \frac{\mu_{ki}^m}{|x_{kj} + \varepsilon_{kj} - v_{ij}|}.$$

From (6),

$$\frac{\partial J_{ij}}{\partial v_{ij}} = -2\sum_{k=1}^{n} w_{ki}(x_{kj} + \varepsilon_{kj} - v_{ij}) = 0.$$

Then, we have

$$v_{ij} = \frac{\sum_{k=1}^{n} w_{ki}(x_{kj} + \varepsilon_{kj})}{\sum_{k=1}^{n} w_{ki}}. \tag{7}$$

Next, we consider the way to obtain ε_{kj}. The procedure is as same as v_{ij}.

Method 1

Step 1. Data is sorted in ascending order in each dimension.

$$v_{1j} - x_{kj}, \ldots, v_{cj} - x_{kj}$$
$$\downarrow \text{Sorting}$$
$$v_{q(1)j} - x_{kj} \leq \ldots \leq v_{q(c)j} - x_{kj}$$

where $q(i)$ is substitution of $(1, \ldots, c)$.

Step 2. We calculate as follows.

$$S = -\frac{1}{2}\sum_{i=1}^{c}(\mu_{ki})^m.$$

Step 3. It starts from $r = 0$ and the following calculations are repeated between $S < 0$.

$$r := r + 1;$$
$$S := S + (\mu_{kq(r)})^m;$$

Step 4. From the above calculation, we obtain

$$\varepsilon_{kj} = \text{sign}(v_{q(r)j} - x_{kj}) \times \min\{|v_{q(r)j} - x_{kj}|, \kappa_{kj}\}. \tag{8}$$

Method 2

From (3), semi-objective function is

$$J_{kj}(\varepsilon_{kj}) = \sum_{i=1}^{c} w_{ki}(x_{kj} + \varepsilon_{kj} - v_{ij})^2. \tag{9}$$

We partially differentiate (9) with respect to ε_{kj} and we have

$$\varepsilon_{kj} = \frac{\sum_{i=1}^{c} w_{ki}(v_{ij} - x_{kj})}{\sum_{i=1}^{c} w_{ki}}. \tag{10}$$

From (1) and (10), we obtain

$$\varepsilon_{kj} = \text{sign}\left(\frac{\sum_{i=1}^{c} w_{ki}(v_{ij} - x_{kj})}{\sum_{i=1}^{c} w_{ki}}\right) \times \min\left\{\left|\frac{\sum_{i=1}^{c} w_{ki}(v_{ij} - x_{kj})}{\sum_{i=1}^{c} w_{ki}}\right|, \kappa_{kj}\right\}. \tag{11}$$

2.2 eFCM for Data with Tolerance That Based on L_1-Norm

The objective function is

$$J_{\text{eFCMT}_{\text{R}}\text{-}L_1} = \sum_{i=1}^{c}\sum_{k=1}^{n} \mu_{ki} d_{ki} + \lambda^{-1} \sum_{i=1}^{c}\sum_{k=1}^{n} \mu_{ki} \log \mu_{ki}. \tag{12}$$

The constraint condition is (2).

The following optimal solution is obtained by using the Lagrange function.

$$\mu_{ki} = \frac{e^{-\lambda d_{ki}}}{\sum_{s=1}^{c} e^{-\lambda d_{ks}}}. \tag{13}$$

Two methods to obtain v_{ij} are considered in the same way of Section **2.1**.

Method 1

From (12), semi-objective function is

$$J_{ij}(v_{ij}) = \sum_{k=1}^{n} \mu_{ki} |x_{kj} + \varepsilon_{kj} - v_{ij}|.$$

Then, v_{ij} is obtained according to the procedure similar to the Section **2.1** by replacing $(\mu_{ki})^m$ and $(\mu_{q(k)i})^m$ with μ_{ki} and $\mu_{q(k)i}$ respectively.

Method 2

From (12), semi-objective function is

$$J_{ij}(v_{ij}) = \sum_{k=1}^{n} w_{ki}(x_{kj} + \varepsilon_{kj} - v_{ij})^2, \tag{14}$$

where

$$w_{ki} = \frac{\mu_{ki}}{|x_{kj} + \varepsilon_{kj} - v_{ij}|}.$$

Similar to Section **2.1**

$$v_{ij} = \frac{\sum_{k=1}^{n} w_{ki}(x_{kj} + \varepsilon_{kj})}{\sum_{k=1}^{n} w_{ki}}. \tag{15}$$

We can get also ε_{kj} Section **2.1**.

Method 1

ε_{kj} is obtained according to the procedure similar to the Section **2.1** by replacing $(\mu_{ki})^m$ and $(\mu_{kq(i)})^m$ with μ_{ki} and $\mu_{kq(i)}$ respectively.

Method 2

(11) is obtained according to the procedure similar to Section **2.1**.

2.3 sFCM for Data with Tolerance That Based on Euclidean Norm

We introduce the tolerance into the objective function of sFCM by Bezdek[1].

$$J_{\text{sFCMT}_\text{R}\text{-}L_2} = \sum_{i=1}^{c}\sum_{k=1}^{n} \mu_{ki}^m \|x_k + \varepsilon_k - v_i\|^2,$$

where

$$\|x_k + \varepsilon_k - v_i\|^2 = \sum_{j=1}^{p}(x_{kj} + \varepsilon_{kj} - v_{ij})^2,$$

under the constraints

$$\sum_{i=1}^{c} \mu_{ki} = 1,\ (\mu_{ki} \geq 0), \tag{16}$$

$$\varepsilon_{kj}^2 \leq \kappa_{kj}^2,\ (\kappa_{kj} > 0). \tag{17}$$

We introduce the following Lagrange function to solve the optimization problem,

$$L_s = \sum_{k=1}^{n}\sum_{i=1}^{c} \mu_{ki}^m \|x_k + \varepsilon_k - v_i\|^2 + \sum_{k=1}^{n} \gamma_k \Big(\sum_{i=1}^{c} \mu_{ki} - 1\Big) + \sum_{k=1}^{n}\sum_{j=1}^{p} \delta_{kj}(\varepsilon_{kj}^2 - \kappa_{kj}^2).$$

From the Kuhn-Tucker condition, the necessary conditions are as follows.

$$\begin{cases} \frac{\partial L_s}{\partial v_{ij}} = 0,\ \frac{\partial L_s}{\partial \mu_{ki}} = 0,\ \frac{\partial L_s}{\partial \varepsilon_{kj}} = 0, \\ \frac{\partial L_s}{\partial \gamma_k} = 0,\ \frac{\partial L_s}{\partial \delta_{kj}} \leq 0, \delta_{kj} \frac{\partial L_s}{\partial \delta_{kj}} = 0, \delta_{kj} \geq 0. \end{cases} \tag{18}$$

From the convexity of $J_{\text{sFCMT}_{\text{R}}\text{-}L_2}$, it is sufficient to consider the case of (18).

For μ_{ki}, from

$$\frac{\partial L_1}{\partial \mu_{ki}} = m\mu_{ki}^{m-1}\|x_k + \varepsilon_k - v_i\|^2 + \gamma_k = 0, \tag{19}$$

we have

$$\mu_{ki} = \left(\frac{-\gamma_k}{m\|x_k + \varepsilon_k - v_i\|^2}\right)^{\frac{1}{m-1}}.$$

In addition, from the constraint condition (16), we have

$$\sum_{l=1}^{c}\left(\frac{-\gamma_k}{m\|x_k + \varepsilon_k - v_l\|^2}\right)^{\frac{1}{m-1}} = 1. \tag{20}$$

From (19) and (20), we have

$$\mu_{ki} = \left(\sum_{l=1}^{c}\left(\frac{\|x_k + \varepsilon_k - v_i\|^2}{\|x_k + \varepsilon_k - v_l\|^2}\right)^{\frac{1}{m-1}}\right)^{-1}. \tag{21}$$

For v_{ij}, from

$$\frac{\partial L_s}{\partial v_{ij}} = -\sum_{k=1}^{n} 2\mu_{ki}^m(x_{kj} + \varepsilon_{kj} - v_{ij}) = 0,$$

we have

$$v_{ij} = \frac{\sum_{k=1}^{n}\mu_{ki}^m(x_{kj} + \varepsilon_{kj})}{\sum_{k=1}^{n}\mu_{ki}^m}. \tag{22}$$

For ε_{kj}, from

$$\frac{\partial L_s}{\partial \varepsilon_{kj}} = \sum_{k=1}^{n} 2\mu_{ki}^m(x_{kj} + \varepsilon_{kj} - v_{ij}) + 2\delta_{kj}\varepsilon_{kj} = 0,$$

we have

$$\varepsilon_{kj} = \frac{-\sum_{i=1}^{c}\mu_{ki}^m(x_{kj} - v_{ij})}{\sum_{i=1}^{c}\mu_{ki}^m + \delta_{kj}}. \tag{23}$$

On the other hand, from

$$\delta_{kj}\frac{\partial L_s}{\partial \delta_{kj}} = \delta_{kj}(\varepsilon_{kj}^2 - \kappa_{kj}^2) = 0,$$

we have to consider two cases, $\delta_{kj} = 0$ and $\varepsilon_{kj}^2 = \kappa_{kj}^2$. First, we consider the case of $\delta_{kj} = 0$. From

$$\frac{\partial L_s}{\partial \varepsilon_{kj}} = \sum_{k=1}^{n} 2\mu_{ki}^m (x_{kj} + \varepsilon_{kj} - v_{ij}) = 0,$$

we have

$$\varepsilon_{kj} = \frac{-\sum_{i=1}^{c} \mu_{ki}^m (x_{kj} - v_{ij})}{\sum_{i=1}^{c} \mu_{ki}^m}. \tag{24}$$

Next, we consider the case of $\varepsilon_{kj}^2 = \kappa_{kj}^2$. From (23) and $\varepsilon_{kj}^2 = \kappa_{kj}^2$, we have

$$\varepsilon_{kj}^2 = \left\{ \frac{-\sum_{i=1}^{c} \mu_{ki}^m (x_{kj} - v_{ij})}{\sum_{i=1}^{c} \mu_{ki}^m + \delta_{kj}} \right\}^2 = \kappa_{kj}^2.$$

Then, we have

$$\sum_{i=1}^{c} \mu_{ki}^m + \delta_{kj} = \pm \frac{|\sum_{i=1}^{c} \mu_{ki}^m (x_{kj} - v_{ij})|}{\kappa_{kj}}.$$

From $\delta_{kj} \geq 0$, the right side is positive. Then, from (23) we get

$$\varepsilon_{kj} = -\frac{\kappa_{kj} \sum_{i=1}^{c} \mu_{ki}^m (x_{kj} - v_{ij})}{|\sum_{i=1}^{c} \mu_{ki}^m (x_{kj} - v_{ij})|}. \tag{25}$$

ε_{kj} which satisfies (24) and (25) is

$$\varepsilon_{kj} = -\alpha_{kj} \sum_{i=1}^{c} \mu_{ki}^m (x_{kj} - v_{ij}), \tag{26}$$

where

$$\alpha_{kj} = \min \left\{ \frac{\kappa_{kj}}{|\sum_{i=1}^{c} \mu_{ki}^m (x_{kj} - v_{ij})|}, \frac{1}{\sum_{i=1}^{c} \mu_{ki}^m} \right\}.$$

2.4 eFCM for Data with Tolerance That Based on Euclidean Norm

The objective function is

$$J_{\text{eFCMT}_\text{R}\text{-}L_2} = \sum_{k=1}^{n} \sum_{i=1}^{c} \mu_{ki} \|x_k + \varepsilon_k - v_i\|^2 + \lambda^{-1} \sum_{k=1}^{n} \sum_{i=1}^{c} \mu_{ki} \log \mu_{ki},$$

under the constraints (16) and (17).

Similar to Section **2.3**, we introduce the following Lagrange function to solve the optimization problem,

$$\begin{aligned} L_e = & \sum_{k=1}^{n} \sum_{i=1}^{c} \mu_{ki} \|x_k + \varepsilon_k - v_i\|^2 + \lambda^{-1} \sum_{k=1}^{n} \sum_{i=1}^{c} \mu_{ki} \log \mu_{ki} \\ & + \sum_{k=1}^{n} \gamma_k \Big(\sum_{i=1}^{c} \mu_{ki} - 1 \Big) + \sum_{k=1}^{n} \sum_{j=1}^{p} \delta_{kj} (\varepsilon_{kj}^2 - \kappa_{kj}^2). \end{aligned}$$

From the Kuhn-Tucker condition, the necessary conditions are as follows.

$$\begin{cases} \frac{\partial L_e}{\partial v_{ij}} = 0,\ \frac{\partial L_e}{\partial \mu_{ki}} = 0,\ \frac{\partial L_e}{\partial \varepsilon_{kj}} = 0, \\ \frac{\partial L_e}{\partial \gamma_k} = 0,\ \frac{\partial L_e}{\partial \delta_{kj}} \leq 0, \delta_{kj} \frac{\partial L_e}{\partial \delta_{kj}} = 0, \delta_{kj} \geq 0. \end{cases} \tag{27}$$

From the convexity of $J_{\text{eFCMT}_\text{R}\text{-}L_2}$, it is sufficient to consider the case of (27).

For μ_{ki}, from

$$\frac{\partial L_e}{\partial \mu_{ki}} = \|x_k + \varepsilon_k - v_i\|^2 + \lambda^{-1}(\log \mu_{ki} + 1) + \gamma_k = 0,$$

we have

$$\mu_{ki} = \frac{e^{-\lambda\|x_k+\varepsilon_k-v_i\|^2}}{\sum_{l=1}^{c} e^{-\lambda\|x_k+\varepsilon_k-v_l\|^2}}. \tag{28}$$

For v_{ij}, we obtain following equation by calculating in the same way of Section **2.3**.

$$v_{ij} = \frac{\sum_{k=1}^{n} \mu_{ki}(x_{kj} + \varepsilon_{kj})}{\sum_{k=1}^{n} \mu_{ki}}. \tag{29}$$

For ε_{kj}, from (14) and

$$\frac{\partial L_e}{\partial \varepsilon_{kj}} = \sum_{k=1}^{n} 2\mu_{ki}(x_{kj} + \varepsilon_{kj} - v_{ij}) + 2\delta_{kj}\varepsilon_{kj} = 0,$$

we have

$$\varepsilon_{kj} = \frac{-\sum_{i=1}^{c} \mu_{ki}(x_{kj} - v_{ij})}{\delta_{kj} + 1}.$$

According to the procedure of Section **2.3**, we obtain

$$\varepsilon_{kj} = -\alpha_{kj}(x_{kj} - \sum_{i=1}^{c} \mu_{ki} v_{ij}), \tag{30}$$

where

$$\alpha_{kj} = \min\left\{ \frac{\kappa_{kj}}{|x_{kj} - \sum_{i=1}^{c} \mu_{ki} v_{ij}|}, 1 \right\}.$$

3 Algorithms

The algorithms derived in the above section are called **sFCMT$_\text{R}$-L_1-1sFCMT$_\text{R}$-L_1-2 eFCMT$_\text{R}$-L_1-1eFCMT$_\text{R}$-L_1-2 sFCMT$_\text{R}$-L_2**and **eFCMT$_\text{R}$-L_2** in turn.

Each algorithm is calculated according to the following procedure. Please refer to Table 1 for the equation of those optimal solutions.

Algorithm

Step 1 Give the values of m and κ_{kj}, and set initial values of E and V.
Step 2 Calculate $U = \mu_{ki}$ by Eq.**A**.
Step 3 Calculate $V = v_{ij}$ by Eq.**B**.
Step 4 Calculate $E = \varepsilon_{kj}$ by Eq.**C**.
Step 5 If (U, E, V) is convergent, stop. Otherwise, go back to **Step 2**.

Table 1. This table shows optimal solutions of each algorithm

Algorithm	Eq.**A**	Eq.**B**	Eq.**C**
sFCMT$_{\mathrm{R}}$-L_1-1	(4)	(5)	(8)
sFCMT$_{\mathrm{R}}$-L_1-2	(4)	(7)	(11)
eFCMT$_{\mathrm{R}}$-L_1-1	(13)	(5)	(8)
eFCMT$_{\mathrm{R}}$-L_1-2	(13)	(15)	(11)
sFCMT$_{\mathrm{R}}$-L_2	(21)	(22)	(26)
eFCMT$_{\mathrm{R}}$-L_2	(28)	(29)	(30)

4 Numerical Examples

In this section, we show some examples of classification by the above-mentioned six algorithms. The classified data set is diagnosis of heart disease[11]. The result of the diagnosis is known. We choose five attributes from 13 ones of original data referring to the advice of a specialist. The number of data is 866 and 560 data contains missing values in some attributes. Please refer to Table 2 for the explanation of each attribute and the number of missing values.

Table 2. The explanation of each attribute and the number of missing values

Attribute	Number of missing values
Resting blood pressure	5
Maximum heart rate achieved	1
ST depression induced by exercise relative to rest	8
The slope of the peak exercise ST segment	255
Number of major vessels colored by flouroscopy	557

To treat missing values as tolerance, we give the average of maximum value and minimum one to the missing one of each attribute, and set the maximum tolerance κ_{kj} on the absolute value of difference between the average and the minimum value.

In all algorithms, the convergence condition is

$$\max_{i,j} |v_{ij} - \bar{v}_{ij}| < 10^{-6},$$

where $\bar{v}_{ij}$is the previous optimal solution. In addition, $m = 2$ in sFCMT$_{\mathrm{R}}$ and $\lambda = 3$ in eFCMT$_{\mathrm{R}}$.

In each algorithm, we give initial cluster centers at random and classify the data set into two clusters. We run this trial 1000 times and show the average of ratio of correctly classified results. Please refer to Table 3 for the results of only 306 data without missing values, Table 4 for the results of the classification by using the algorithms proposed in this paper and Table 5 for the results of the

Table 3. The results of classifying only 306 data without missing values

Algorithm	The average of ratio correctly classified
sFCM-L_1-1	70.0
sFCM-L_1-2	71.9
eFCM-L_1-1	74.1
eFCM-L_1-2	74.8
sFCM-L_2	75.2
eFCM-L_2	69.2

Table 4. The results of the classification by using the proposed algorithms in this paper

Algorithm	The average of ratio correctly classified
sFCMT$_{\mathrm{R}}$-L_1-1	68.6
sFCMT$_{\mathrm{R}}$-L_1-2	67.4
eFCMT$_{\mathrm{R}}$-L_1-1	66.2
eFCMT$_{\mathrm{R}}$-L_1-2	62.4
sFCMT$_{\mathrm{R}}$-L_2	73.4
eFCMT$_{\mathrm{R}}$-L_2	61.3

Table 5. The results of the classification by using the algorithms which treat missing value as interval data and use nearest neighbor distance to calculate dissimilarity

Algorithm	The average of ratio correctly classified
sFCM-L_1-1	69.0
sFCM-L_1-2	68.9
eFCM-L_1-1	69.2
eFCM-L_1-2	68.6
sFCM-L_2	67.2
eFCM-L_2	67.6

classification by using the algorithms which treat missing value as interval data and uses nearest neighbor distance to calculate the dissimilarity.

To compare the results for all data by the proposal algorithms(Table 4) with the results for only data without missing values(Table 3), the latter is a little better than the former. However, this is very natural. From these examples, we can not find significant difference between the algorithms, using the tolerance and nearest neighbor distance. The important point is that ε is calculated only by the proposed algorithms. The meaning of ε depends on the data set.

5 Conclusion

In this paper, we considered the optimization problems for data with tolerance and solved the optimal solutions. Using the results, we have constructed new

six algorithms. Moreover, we shown the usefulness of the proposed algorithms through some numerical examples.

In these algorithms, data with tolerance is not regarded as interval data. The reason is that the algorithms is more appropriate because we can use the former dissimilarities based on normal distance in the frame of the optimization. Moreover, we can use the proposed algorithms for the data with tolerance defined as hyper-rectangle in more cases than the algorithms for the data with tolerance defined as hyper-sphere [6,7] because more certainties should be represented as hyper-rectangle than hyper-sphere.

In the forthcoming paper, we will consider to apply the concept of tolerance to regression analysis and support vector machine.

References

1. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum, New York (1981)
2. Takata, O., Miyamoto, S.: Fuzzy clustering of Data with Interval Uncertainties. Journal of Japan Society for Fuzzy Theory and Systems 12(5), 686–695, 2000 (in Japanese)
3. Sato-Ilic, M., Oshima, J.: On Weighted Principal Component Analysis for Interval-Valued Data and Its Dynamic Feature. International Journal of Innovative Computing, Information and Control 2(1), 69–82 (2006)
4. Endo, Y.: Clustering Algorithm Using Covariance for Fuzzy Data. In: Proc. 1998 International Symposium on Nonlinear Theory and Its Applications, pp. 511–514 (1998-99)
5. Endo, Y., Horiuchi, K.: On Clustering Algorithm for Fuzzy Data. In: Proc. 1997 International Symposium on Nonlinear Theory and Its Applications, pp. 381–384 (1997.11)
6. Murata, R., Endo, Y., Haruyama, H., Miyamoto, S.: On Fuzzy c-Means for Data with Tolerance. Journal of Advanced Computational Intelligence and Intelligent Informatics 10(5), 673–681 (2006)
7. Toyoda, H.: L_1-Norm based Fuzzy Clustering for Data with Tolerance, Graduation thesis, College of Engineering Systems, University of Tsukuba, 2005 (in Japanese)
8. Miyamoto, S., Mukaido, M.: Fuzzy c-means as a regularization and maximum entropy approach. In: Proc. of the 7th International Fuzzy Systems Association World Congress (IFSA'97), Prague, Chech, June 25-30, 1997, vol. 2, pp. 86–92 (1997)
9. Jajuga, K.: L_1-norm based fuzzy clustering. Fuzzy Sets and Systems 39, 43–50 (1991)
10. Koga, T.: Clustering Algorithm based on L_1-norm space, Master's thesis, Graduate School of Systems and Information Engineering, University of Tsukuba, 2002 (in Japanese)
11. UCI Machine Learning Databases: `http://www.ics.uci.edu/~mlearn/databases/heart-disease/`

Kernel Functions Based on Fuzzy Neighborhoods and Agglomerative Clustering

Sadaaki Miyamoto[1] and Yuichi Kawasaki[2]

[1] Department of Risk Engineering
School of Systems and Information Engineering
University of Tsukuba, Ibaraki 305-8573, Japan
miyamoto@risk.tsukuba.ac.jp
[2] Graduate School of Systems and Information Engineering
University of Tsukuba, Ibaraki 305-8573, Japan
ttyrkwsk@soft.risk.tsukuba.ac.jp

Abstract. A fuzzy neighborhood model for analyzing information systems having topological structures on occurrences of keywords is proposed and associated kernel functions are studied. Sufficient conditions when a neighborhood defines a kernel are derived. Accordingly, agglomerative clustering algorithms are applicable which employ kernel functions. An illustrative example is given.

1 Introduction

Text mining and web information analysis are hot topics of today; yet existing methods are insufficient, and hence a new theory for these applications should be developed. From the viewpoint of rough sets [11,12], the classical framework of classifications of the universal set is frequently too strict, and more general structure of the topology should be considered, as texts and web information have natural topologies.

We have proposed a model of fuzzy neighborhoods [9] which partly generalizes non-standard rough sets [16] using fuzzy relations. Another feature of this model is to analyze a set of terms of which occurrences are dispersed on an universal set with a natural topology. Accordingly dissimilarity measures are defined on the set of terms and classification and clustering of terms have been considered. A drawback of this model is that such a dissimilarity measure is a weaker mathematical structure than the Euclidean metric in which not only the distance but also the cosine between two objects are defined.

In this paper we proceed to introduce an inner product space into the neighborhood model using the high-dimensional feature space used in support vector machines [15]. In particular the idea of the convolution kernel [4] is employed. A type of data is focused upon that is a sequence of occurrences of terms. Two sufficient conditions that the fuzzy neighborhoods provide kernel functions will be given. Namely, a theorem by Pólya [13] can be used, and moreover a fuzzy equivalence relation defines another kernel. Accordingly agglomerative clustering algorithms are applicable including the centroid method and the Ward method [3,5],

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 249–260, 2007.

which assumes the use of a Euclidean distance. Finally an illustrative example is given to show the present algorithm correctly works.

2 Term Space and Fuzzy Neighborhood Space

Throughout this paper, we assume $A(x)$ means the membership value of fuzzy set A at x instead of the classical symbol $\mu_A(x)$.

The fuzzy neighborhood model basically consists of the quadruple

$$< T, O, R, N > \tag{1}$$

in which T is called a term set in which elements are denoted by $t, t', t_i, \ldots \in T$; O is called an occurrence space in which elements are denoted by $o, o', o_i, \ldots \in O$. T is generally a finite set, while O can be either finite or infinite. R is a fuzzy relation on $T \times O$, while N is a fuzzy relation on $O \times O$. We moreover define a family of fuzzy sets $N[o]$ of the same symbol:

$$N(o, o') = N[o](o'), \quad \forall o' \in O. \tag{2}$$

That is, $N[o]$ is a fuzzy set that is dependent on o, which is defined by the above equation. The fuzzy set $N[o]$ is called a *fuzzy neighborhood* of $o \in O$.

In applications, the set T is a set of keywords which we wish to classify or divide into clusters. In real worlds, the keywords may occur many times in a text, or distributed on web pages. Hence keyword occurrences are represented by $o, o' \ldots \in O$. When o means a keyword t, $R(t, o) = 1$. However, there are similar keywords and hence generally $0 \leq R(t, o) \leq 1$. When an occurrence o does not correspond to any keyword t, $R(t, o) = 0$. We assume, for each $t \in T$, there exists $o \in O$ such that $R(t, o) > 0$. We hereafter use the word of a *term* instead of a *keyword*.

Basically we assume $N[o]$, a neighborhood of occurrence o, is given. In applications, however, a distance may be given instead of a neighborhood. When a distance $d(o, o')$ is defined on O, we can define $N[o]$ by the next procedure.

(i) Let $f\colon [0, \infty) \rightarrow [0, 1]$ be a strictly monotonically decreasing function such that $f(0) = 1$ and

$$\lim_{x \rightarrow +\infty} f(x) = 0. \tag{3}$$

(ii) Define $N[o]$ by

$$N[o](o') = f((d(o, o')). \tag{4}$$

Notice that the relation $N(o, o')$ defined by (4) satisfies the symmetry

$$N(o, o') = N(o', o). \tag{5}$$

Although there are other ways to define directly the neighborhoods, we assume the symmetry throughout this paper.

We moreover state additional assumptions.

(I) Reflexivity: $N(o,o) = 1$ for all $o \in O$.
(II) Finiteness: For an arbitrary $o \in O$, $|\{o' : N(o,o') > 0\}| < +\infty$, i.e., the number of elements which have nonzero membership values are finite.

The interpretation of $N[o]$ is straightforward, If $N[o](o') > N[o](o'')$, then o' is nearer to o than o''. If $N[o](o') = 0$, then the relation of o to o' is neglected by the model.

2.1 Text Mining: Term Relations in Text Sets

Although there is an important application area of web information, we focus upon documents as a sequence of terms. Let the set of documents be $\mathcal{D} = \{d_1, d_2, \ldots, d_n\}$. A document d consists of a sequence of occurrences. For simplicity, suppose an occurrence corresponds to a unique term. We handle a sequence of occurrences, and accordingly we define $Sqnc(d)$ is the sequence of occurrences. For example assume $Sqnc(d) = abcde$ and term of a, b, d is t; term of c and e is t'. Then $Sqnc(d) = ttt'tt'$ using the term symbols.

From technical reason we define concatenation of two document sequences by $Sqnc(d)|Sqnc(d')$. Thus if

$$Sqnc(d) = abcd, \qquad Sqnc(d') = vwxyz,$$

then

$$Sqnc(d)|Sqnc(d') = abcdvwxyz.$$

The whole sequence X of the document set $\mathcal{D}$ is

$$X = Sqnc(d_1)|Sqnc(d_2)|\cdots|Sqnc(d_n).$$

A natural distance D is defined:

$$D(a,b) = \{ \text{ number of term occurrences between } a \text{ and } b \ \} + 1 \tag{6}$$

For the above d and d', $D(v,w) = 1$ and $D(v,z) = 4$. Thus for this distance the fuzzy neighborhood is naturally defined. The next two are typical examples.

Crisp and fuzzy neighborhoods $N_{C_K}[a]$ and $N_{F_K^\beta}[a]$:

$$N_{C_K}[a](x) = \begin{cases} 1 & (D(a,x) \le K), \\ 0 & (D(a,x) > K). \end{cases} \tag{7}$$

$$N_{F_K^\beta}[a](x) = \begin{cases} 1 - \{D(a,x)/K\}^\beta & (D(a,x) \le K), \\ 0 & (D(a,x) > K). \end{cases} \tag{8}$$

The meanings of these neighborhoods are clear.

3 Inner Product Space Derived from Fuzzy Neighborhood

Generally a system of fuzzy neighborhoods defines a weak topological structure than the standard Euclidean space. In other words, we cannot construct a strong mathematical model whereby various theoretical tools and algorithms are applied. One reason why the Euclidean space has good theoretical properties is that it has the inner product. Hence if we derive an inner product on the space of terms using fuzzy neighborhoods, we can expect most mathematical tools of data analysis are applicable. Therefore the main question herein is when and how we can derive an inner product in the present framework. The key idea is the convolution kernel [4].

3.1 A Proximity Measure

Let us define a proximity measure $p(t, t')$ between two terms $t, t' \in T$ and a normalized measure $s(t, t')$ associated with the former.

$$p(t, t') = \sum_{a \in O} \sum_{b \in O} R(t, a) N(a, b) R(t', b) \tag{9}$$

$$s(t, t') = \frac{p(t, t')}{\sqrt{p(t, t) p(t', t')}}. \tag{10}$$

What we will show in this paper is when and how the above measure $p(t, t')$ becomes an inner product and accordingly $s(t, t')$ is a cosine correlation [3] associated with $p(t, t')$.

3.2 Vector Space Derived from Terms

Let us suppose for the moment that $p(t, t')$ is a positive definite function, i.e., the eigenvalues of matrix $P = (p(t, t'))$ are all positive, or in other words,

$$\sum_{t, t' \in T} z_t z_{t'} p(t, t') > 0, \quad \text{for all } (z_t) \neq 0.$$

Then we can define a vector space $\boldsymbol{R}^{|T|}$ with the inner product

$$\langle z, z' \rangle_P = z^t P z, \quad z, z' \in \boldsymbol{R}^{|T|}.$$

Such a positive definite matrix provides an inner product of a vector space. In such a case the matrix or a function $p(t, t')$ is called a kernel function.

The correspondence between a term t and z_t, the t-th component of z can be explained in terms of real-valued bags (or multisets) [8]. Thus, z_t implies the weight on the term t which can both be positive or negative. Such an interpretation of a term space is not new, since the vector space method [14] is based on such an idea.

Moreover, $p(t,t')$ is called positive semi-definite, if and only if all the eigenvalues of P are nonnegative, or in other words,

$$\sum_{t,t' \in T} z_t z_{t'} p(t,t') \geq 0.$$

In applications, the difference between the positive definite matrix and the positive semi-definite matrix is not very important, since a positive semi-definite matrix can be approximated by a positive definite matrix using the regularization $P + \varepsilon I$ $(\varepsilon > 0)$. Note

$$P + \varepsilon I \to P \quad (\varepsilon \to 0).$$

We therefore do not strongly distinguish the positive definiteness and positive semi-definiteness below.

3.3 Kernel Functions on Terms and Fuzzy Neighborhoods

Let us consider sufficient conditions when a neighborhood defines a kernel function.

To begin with, consider a simple case when $T = O$ and $R(t,a) = 1$ if and only if $t = a$. That is, an occurrence itself is a term. In this case (9) reduces to $p(t,t') = N(t,t')$ and hence the necessary and sufficient condition such that $p(t,t')$ is positive definite is $N(t,t')$ is positive definite.

We proceed to the general case of (9).

Proposition 1. *A sufficient condition such that $p(t,t')$ is positive definite is that $N(a,b)$ is positive definite for all $a, b \in O$.*

Proof. Let $y(a) = \sum_t z_t R(t,a)$. Then, from

$$\begin{aligned} z^t P z &= \sum_{t,t'} z_t z_{t'} \sum_{a,b} R(t,a) N(a,b) R(t',b) \\ &= \sum_{a,b} \sum_{t} z_t R(t,a) \sum_{t'} z_{t'} R(t',b) N(a,b) \\ &= \sum_{a,b} y(a) y(b) N(a,b), \end{aligned}$$

we have

$$\sum_{a,b} y(a) y(b) N(a,b) \geq 0 \implies z^t P z \geq 0.$$

□

We also have

Proposition 2. *If $p(t,t')$ is positive definite, then $s(t,t')$ defined by (10) is also positive definite.*

Proof. Let $\zeta_t = z_t/\sqrt{p(t,t)}$ and note $p(t,t) > 0$ from the positive definiteness of $p(t,t')$. Then,

$$z^t S z = \sum_{t,t'} z_t z_{t'} s(t,t') = \sum_{t,t'} \zeta_t \zeta_{t'} p(t,t') \geq 0,$$

which shows that $s(t,t')$ is positive definite. □

Hence we should study when $N(a,b)$ is positive definite.

3.4 A Kernel Function for Text Mining

Let us consider the term sequence X with the natural distance $D(a,b)$. In other words, the set O of occurrences is the term sequence X. The fuzzy neighborhood is defined:

$$N(a,b) = N[a](b) = f(D(a,b))$$

using (4). We have the following sufficient condition.

Proposition 3. *A sufficient condition such that $N(a,b)$ is positive definite is that the function f is convex on $[0,+\infty)$ and $f(x) \to 0$ as $x \to +\infty$.*

Proof. Take an arbitrary $c \in O$. The conclusion is immediately obtained from Pólya's theorem [13] which states that

$$\sum_{a,b} z_a z_b f(|x(a) - x(b)|) \geq 0$$

when f is convex on $[0,+\infty)$ and $f(x) \to 0$ as $x \to +\infty$. Note that $x(a)$ is the real value defined by $x(a) = D(a,c)$ when a is the left hand of c; $x(a) = -D(a,c)$ when a is right hand of c. □

For example, the fuzzy neighborhood $N_{F_K^\beta}[a](x)$ of (8) is convex if $0 < \beta \leq 1$, while the crisp neighborhood (7) is not guaranteed to provide a positive definite kernel.

Note 1. A function $g\colon \boldsymbol{R}^n \to \boldsymbol{R}$ is said to be convex when

$$g(\lambda x + (1-\lambda)y) \leq \lambda g(x) + (1-\lambda) g(y), \quad \forall x, y \in \boldsymbol{R}^n \text{ and } \forall \lambda \in [0,1].$$

3.5 Fuzzy Rough Approximation and a Kernel Function

A more interesting result for researchers in rough sets is that relations for fuzzy rough approximations [1] provide kernel functions. A fuzzy rough approximation here is defined by a fuzzy equivalence relation on O. That is, $N(a,b)$ is assumed to satisfy the transitivity

$$N(a,b) \geq \min\{N(a,c), N(c,b)\}, \quad \forall c \in O \tag{11}$$

in addition to the reflexivity $N(a,a) = 1$ and the symmetry $N(a,b) = N(b,a)$. Then it is well-known that for arbitrarily fixed $\alpha \in (0,1]$, the subset

$$[N(a,\cdot)]_\alpha = \{b \in O : N(a,b) \geq \alpha\}$$

forms a partition of O: either one of $[N(a,\cdot)]_\alpha = [N(b,\cdot)]_\alpha$ or $[N(a,\cdot)]_\alpha \cap N(b,\cdot)]_\alpha = \emptyset$ holds.

We now have the next proposition.

Proposition 4. *Assume that $N(a,b)$ is a fuzzy equivalence relation. Then $N(a,b)$ and also $p(t,t')$ are positive definite kernel functions.*

Proof. To prove this proposition, we consider a partition matrix $U = (u_{ij})$. Namely, an $n \times n$ real matrix U is called a partition matrix iff there exists a partition $K_1, \ldots, K_c$ of $\mathbf{n} = \{1, 2, \ldots, n\}$ (i.e., $\bigcup_j K_j = \mathbf{n}$ and $K_i \cap K_j = \emptyset$, for $i \neq j$) such that

$$u_{ij} = 1, \quad \forall i, j \in K_h$$

for some h and

$$u_{ij} = 0, \quad \forall i \in K_\ell,\ j \in K_h$$

for $h \neq \ell$.

It should be noted here that a partition matrix U is positive semi-definite, and it is positive definite if and only if U is identity matrix ($U = I$), since

$$x^t U x = \sum_{i=1}^{c} \left(\sum_{x_j \in K_i} x_j \right)^2 .$$

The proof of the proposition is now straightforward. We assume that O is a finite set for simplicity. Then an equivalence relation is represented by a partition matrix. Moreover a fuzzy equivalence relation F is represented by a finite collection $U_1, \ldots, U_k$ of partition matrix and positive $\beta_1, \ldots, \beta_k$:

$$F = \sum_{j=1}^{k} \beta_j U_j .$$

Consequently we have

$$x^t F x = \sum_{j=1}^{k} \beta_j x^t U_j x \geq 0.$$

Hence $N(a,b)$ is positive definite. From Proposition 1, $p(t,t')$ is also positive definite. The proposition is thus proved. □

It has been known that a fuzzy equivalence relation gives a hierarchical classification [3,5]. Conversely, given a hierarchical classification represented by a tree, we can define a fuzzy equivalence relation by putting a real number to each node of the tree.

Another question is how to obtain a fuzzy equivalence relation, since the condition of transitivity (11) seems strict. A natural answer is to perform the single link method of agglomerative clustering [3], which is equivalent to connected components of a fuzzy graph [5]. In terms of dissimilarity measure, the transitivity is called an ultrametric [3]. To summarize, to obtain a fuzzy equivalence relation is not very difficult, but the single link clustering is necessary if we do not have a predefined categorical structure.

4 Application to Agglomerative Clustering

We show the general procedure **AHC** of agglomerative clustering in which $\mathcal{G} = \{G_1, \ldots, G_C\}$ is a family of clusters $G_1, \ldots, G_C$ which forms a partition of T. Note also that $d(G, G')$ is a dissimilarity measure between two clusters.

Algorithm AHC (Agglomerative Hierarchical Clustering)
AHC0. Put the number of clusters $C = |T|$; initialize clusters:
$G_i = \{t_i\}$, $i = 1, \ldots, C$; $d(G_i, G_j) = d(t_i, t_j)$, all i, j.
AHC1. Calculate $(G, G') = \arg \min_{1 \leq i,j \leq C, i \neq j} d(G_i, G_j)$.
Put $\hat{G} = G \cup G'$. Remove G, G' from $\mathcal{G}$ and add $\hat{G}$ to $\mathcal{G}$.
AHC2. Let $C = C - 1$. If $C = 1$, output the dendrogram and stop.
Else update $d(\hat{G}, G'')$, $\forall G'' \in \mathcal{G}$, and go back to **AHC1**.
End of AHC.

Among different ways to update the dissimilarity measures, well-known techniques are the single-link, the complete link, the average-link, the centroid method, and the Ward method [3,5]. It is immediate to see that the single link, the complete link, and the average link are applicable to $p(t, t')$ and $s(t, t')$, while the centroid method and the Ward method require the assumption of a Euclidean space. It has been shown, however, that these two methods can also be used when a kernel function is employed, and moreover that the updating formulas are just the same as the ordinary formulas [2]. Hence the only difference in the nernel-based agglomerative clustering is the initial values $d(t, t')$. We can take either $p(t, t')$ or $s(t, t')$ as the inner product, but let us take $s(t, t')$ here. For the centroid method,

$$d(t, t') = 2 - 2s(t, t')$$

and for the Ward method,

$$d(t, t') = 1 - s(t, t').$$

4.1 An Illustrative Example

A simple example is shown to see that the algorithm correctly works. We use the Ward method with two fuzzy neighborhoods on a simple sequence of term occurrences

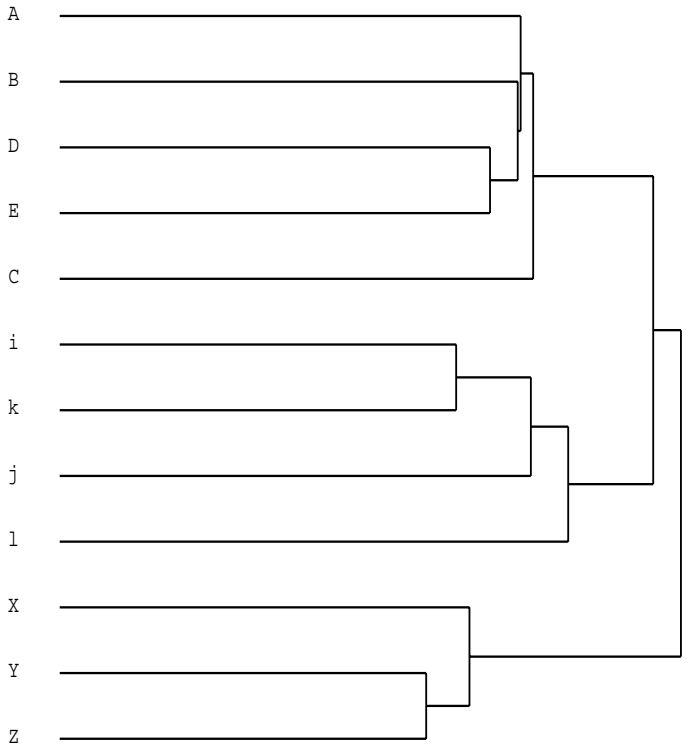


Fig. 1. Dendrogram from the letter sequence using the Ward method with $s(t, t')$ and the kernel N_1

```
A,B,A,C,i,j,j,l,A,D,E,D,E;Y,Z,A,E,B,D,j,k,i,k;
B,C,A,C;E,X,Y,X,Z,B,E,C,D;i,l,k,l,A,D,B,A,C,i,j;
j,l;A,D,B,D,E,A,A,Z,Z,Y,Y,Z;A,E,B,D;
j,k,i,k,B,C,A,E,X,Y,X,Z;B,E,C,D,i,l,k,l
```

in which a letter stands for an occurrence separated by a comma(`,`), while the semicolon(`;`) implies separation of documents and hence no neighborhood can intersect the semicolon. The correct answer should be three clusters of $\{\mathtt{A},\mathtt{B},\mathtt{C},\mathtt{D},\mathtt{E}\}$, $\{\mathtt{i},\mathtt{j},\mathtt{k},\mathtt{l}\}$, $\{\mathtt{X},\mathtt{Y},\mathtt{Z}\}$.

The two fuzzy neighborhoods are

$$N_1(x) = 2^{-|x|} \tag{12}$$

$$N_2(x) = \begin{cases} 1 - |x|/20 & (|x| \leq 20) \\ 0 & (|x| > 20) \end{cases} \tag{13}$$

It is immediate to see that these two neighborhoods satisfy the condition in Proposition 3.

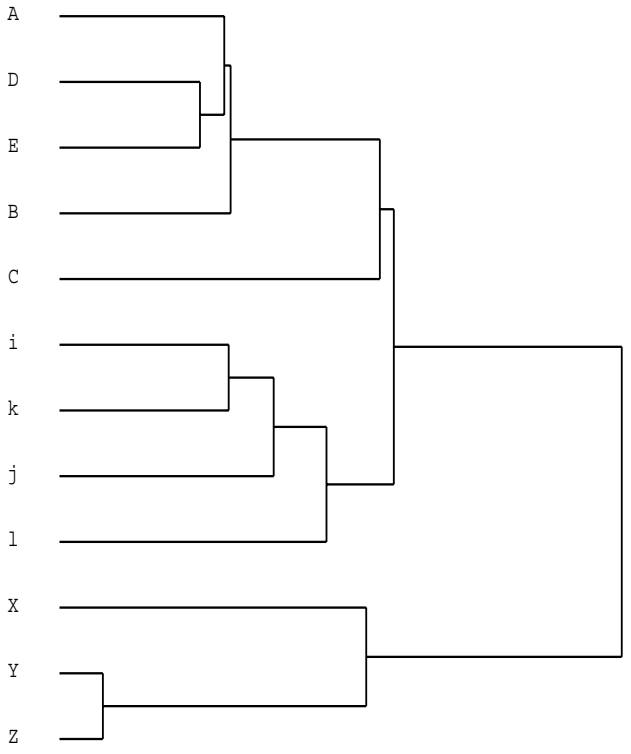


Fig. 2. Dendrogram from the letter sequence using the Ward method with $s(t, t')$ and the kernel N_2

We show the two dendrograms by the Ward method using $s(t, t')$ with the kernels N_1 and N_2, respectively, in Figures 1 and 2. We see the correct results in these dendrograms.

5 Conclusions

In this paper a model of fuzzy neighborhoods with application to text mining is shown and associated kernel functions are proposed, whereby the term set can be interpreted as a vector space with an inner product.

Concerning the relation between the present theory and rough sets, a neighborhood system [10] is proposed by us that assumes a system of subsets, whereby a family of kernel functions on generalized rough sets is considered, which is different from the kernels herein.

Although we have mainly shown theoretical properties in this paper without any real example, the present results imply that a number of existing methods in data analysis are applicable to the present model, since an inner product space

has good mathematical properties. Namely, after establishing kernel functions, we can apply many methods of classification and clustering such as the support vector machines [15] and kernel fuzzy c-means [6,7]. In this sense, the kernel method proposed here has broad area of applications.

From theoretical viewpoint, an important point is that several different theories are put together in the present framework, i.e., neighborhood systems, classification and fuzzy classification [5], positive definite functions, agglomerative clustering, and so on.

When different theories meet together in such a way, we have an ample research possibility in near future. The present method is typical in this sense.

Acknowledgment

This study has partly been supported by the Grant-in-Aid for Scientific Research,No.16650044, Japan Ministry of Education, Culture, Sports, Science and Technology, and No.16300065, Japan Society for the Promotion of Science.

References

1. Dubois, D., Prade, H.: Rough fuzzy sets and fuzzy rough sets. Int. J. General Systems 17, 191–209 (1990)
2. Endo, Y., Haruyama, H., Okubo, T.: On Some Hierarchical Clustering Algorithms Using Kernel Functions. In: Proc. of FUZZ-IEEE, Budapest, Hungary, 2004, pp. 1–6 (CD-ROM Proc.) (2004)
3. Everitt, B.S.: Cluster Analysis, 3rd edn. Arnold, London (1993)
4. Haussler, D.: Convolution kernels on discrete structures, Technical Report UCSC-CRL-99-10, UC Santa Cruz (1999)
5. Miyamoto, S.: Fuzzy Sets in Information Retrieval and Cluster Analysis. Kluwer, Dordrecht (1990)
6. Miyamoto, S., Suizu, D.: Fuzzy c-means clustering using transformations into high dimensional spaces. In: Proc. of FSKD'02. 1st International Conference on Fuzzy Systems and Knowledge Discovery, Singapore, November 18-22, 2002, vol. 2, pp. 656–660 (2002)
7. Miyamoto, S., Suizu, D.: Fuzzy c-means clustering using kernel functions in support vector machines. Journal of Advanced Computational Intelligence and Intelligent Informatics 7(1), 25–30 (2003)
8. Miyamoto, S.: Generalizations of multisets and rough approximations. International Journal of Intelligent Systems 19(7), 639–652 (2004)
9. Miyamoto, S., Hayakawa, S.: A Fuzzy Neighborhood Model for Clustering, Classification, and Approximations. In: Greco, S., Hata, Y., Hirano, S., Inuiguchi, M., Miyamoto, S., Nguyen, H.S., Słowiński, R. (eds.) RSCTC 2006. LNCS (LNAI), vol. 4259, pp. 882–890. Springer, Heidelberg (2006)
10. Miyamoto, S.: Neighborhood Systems for Rough Approximations and Data Clustering Algorithms. In: Proc. of ISME, March 10-12, Kitakyushu, Japan, pp. 1–6 (CD-ROM) (2007)
11. Pawlak, Z.: Rough sets. International Journal of Computer and Information Sciences 11, 341–356 (1982)

12. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data. Kluwer, Dordrecht (1991)
13. Pólya, G.: Remarks on characteristic functions. In: Proc. First Berkeley Conf. on Math. Stat. and Prob. pp. 115–123. Univ. of Calf Press, Berkeley (1949)
14. Salton, G., McGill, M.J.: Introduction to Modern Information Retrieval. McGraw-Hill, New York (1983)
15. Vapnik, V.N.: Statistical Learning Theory. Wiley, Chichester (1998)
16. Yao, Y.Y., Wong, S.K.M., Lin, T.Y.: A review of rough set models. In: Lin, T.Y., Cercone, N. (eds.) Rough Sets and Data Mining: Analysis of Imprecise Data, pp. 47–75. Kluwer, Boston (1997)

c-Means Clustering on the Multinomial Manifold

Ryo Inokuchi[1] and Sadaaki Miyamoto[2]

[1] Doctoral Program in Risk Engineering, University of Tsukuba
Ibaraki 305-8573, Japan
inokuchi@soft.risk.tsukuba.ac.jp
[2] Department of Risk Engineering, University of Tsukuba
Ibaraki 305-8573, Japan
miyamoto@risk.tsukuba.ac.jp

Abstract. In this paper, we discuss *c*-means clustering algorithms on the multinomial manifold. Data forms a Riemannian manifold with the Fisher information metric via the probabilistic mapping from datum to a probability distribution. For discrete data, the statistical manifold of the multinomial distribution is appropriate. In general, The euclidean distance is not appropriate on the manifold because the parameter space of the distribution is not flat. We apply the Kullback-Leibler (KL) divergence or the Hellinger distance as approximations of the geodesic distance to hard *c*-means and fuzzy *c*-means.

1 Introduction

Data mining for discrete data, such as graph, text, and biological sequences, is significant for real-world applications. In the discrete setting, the representation of data is crucial. It is not appropriate to use raw data directly because data are very sparse and very high-dimensional.

Recently the probabilistic mapping from datum x to a probability distribution $p(x)$ have been used in [1]. It is a suitable representation of discrete data. For documents, "bag of words" representation is that a document is described as a multinomial distribution (or histogram) of words.

A family of probability distributions forms a Riemannian manifold [2]. The structure of a manifold is given by the Fisher information metric. For the multinomial family, all probability vectors are on m-simplex. The Riemannian manifold is usually "curved" while the Euclidean space is "flat". Therefore, the squared Euclidean distance is not suitable as dissimilarity when we apply the usual *c*-means (K-means) to a set of probability distribution. The geodesic distance by the Fisher information metric or its approximation is needed to clustering on the manifold.

In this paper, we discuss *c*-means clustering on the statistical manifold of the multinomial distribution. It is difficult to apply directly the geodesic distance as a dissimilarity. Instead, we apply the Kullback-Leibler (KL) divergence to hard *c*-means and fuzzy *c*-means. In addition, we discuss the use of the Hellinger distance.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 261–268, 2007.

2 The Multinomial Manifold

We now present a brief introduction of the multinomial manifold. For further details, see [2,3].

The multinomial manifold is the parameter space of the multinomial distribution

$$\mathbb{P}^m = \Big\{ x \in \mathbb{R}^{m+1} : \forall j,\ x_j\ > 0, \quad \sum_{j=1}^{m+1} x_j = 1 \Big\}. \tag{1}$$

The structure of a manifold is given by the Fisher information metric,

$$g_\theta(u,v) = \sum_{i,j} u_i v_j E_{p_x}\Big(\frac{\partial^2 \log p_x(y)}{\partial x_i \partial x_j} \Big) \tag{2}$$

$$= \sum_{j=1}^{m+1} \frac{u_j v_j}{x_j}, \tag{3}$$

where $u, v \in T_x\mathbb{P}^m$ are tangent vectors to $\mathbb{P}^m$ at x represented in the standard basis of $\mathbb{R}^{m+1}$. The inner product (3) defines the structure of the manifold, such as distance, angle, curvature on $\mathbb{P}^m$. It is quite different from the inner product on the Euclidean space.

It is well-known fact the multinomial manifold is isometric to the positive m-sphere

$$\mathbb{S}_+^m = \Big\{ x \in \mathbb{R}^{m+1} : \forall j,\ x_j\ > 0, \quad \sum_{j=1}^{m+1} x_j^2 = 1 \Big\}, \tag{4}$$

through the diffeomorphism $F : \mathbb{P}^m \to \mathbb{S}_+^m$,

$$F(x) = (\sqrt{x_1}, \ldots, \sqrt{x_{m+1}}). \tag{5}$$

Therefore the geodesic distance between $x, x' \in \mathbb{P}^m$ can be computed as the geodesic distance between $F(x), F(x') \in \mathbb{S}_+^m$, the shortest curve connecting $F(x)$ and $F(x')$,

$$D_G(x,x') = \arccos\Big(\sum_{j=1}^{m+1} \sqrt{x_j x'_j} \Big). \tag{6}$$

The direct use of the geodesic distance is much difficult to analyze. We introduce another familiar dissimilarity measures. The Kullback-Leibler (KL) divergence have been frequently used to measure the distance between probability distributions. For the multinomial family, KL-divergence is defined by

$$D_{KL}(x||x') = \sum_{j=1}^{m+1} x_j \log \frac{x_j}{x'_j}. \tag{7}$$

The Hellinger distance have been also used.

$$D_H(x, x') = \sqrt{\sum_{j=1}^{m+1} \left(\sqrt{x_j} - \sqrt{x'_j}\right)^2} \tag{8}$$

3 Hard c-Means Based on KL-Divergence

Probability vectors to be clustered are denoted by $x_k = (x_{k1}, \ldots, x_{k,m+1}) \in \mathbb{P}^{m+1}$, $k = 1, \ldots, n$, Cluster centers are $v_i = (v_{i1}, \ldots, v_{1,m+1})^T$, $i = 1, \ldots, c$, where c is the number of clusters. An abbreviated symbol $V = (v_1, \ldots, v_c)$ is used for the whole collection of cluster centers. The matrix $U = (u_{ik})$, $(i = 1, \ldots, c, k = 1, \ldots, n)$ is used as usual, where u_{ik} means the degree of belongingness of object x_k to cluster i.

As is well-known, hard c-means clustering is based on the optimization of an objective function.

$$J = \sum_{k=1}^{n} \sum_{i=1}^{c} u_{ik} D_{KL}(x_k || v_i) \tag{9}$$

Note that x_k and v_i are probability vectors defined by

$$\sum_{j=1}^{m+1} x_{kj} = \sum_{j=1}^{m+1} v_{ij} = 1. \tag{10}$$

The constraints is assumed for U,

$$M = \{\, U = (u_{ik}) \,:\, \sum_{i=1}^{c} u_{ik} = 1,\ u_{jk} \in \{0, 1\},\ \forall j, k \,\}. \tag{11}$$

A cluster assignment of x_k is the same as a usual hard c-means,

$$l = \arg \min_{1 \le i \le c} D_{KL}(x_k || v_i). \tag{12}$$

Cluster centers are derived from the Lagrange multiplier method with the constraint.

$$L = J + \sum_{i=1}^{c} \mu_i (\sum_{j=1}^{m+1} v_{ij} - 1)$$

$$\frac{\partial L}{\partial v_{ij}} = -\sum_{k=1}^{n} \frac{u_{ik} x_{kj}}{v_{ij}} + \mu_i = 0$$

As a result, cluster centers are calculated in the same way as a usual hard c-means.

$$v_{ij} = \frac{\sum_{k=1}^{n} u_{ik} x_{kj}}{\sum_{k=1}^{n} u_{ik}} \tag{13}$$

Algorithm HCM. (Hard c-means)
HCM0. Set an initial value $\bar{V}, \bar{\alpha}, \bar{S}$ for V, α, S.
HCM1. Find optimal solution of J with respect to U while other variables are fixed: put

$$\bar{U} = \arg\min_{U \in M} J(U, \bar{V}).$$

HCM2. Find optimal solution of J with respect to V while other variables are fixed: put

$$\bar{V} = \arg\min_{V} J(\bar{U}, V).$$

HCM3. If the solution $(\bar{U}, \bar{V})$ is convergent, stop; else go to **HCM1**.
End of HCM.

4 Fuzzy c-Means Based on KL-Divergence

Fuzzy c-means[4] obtains a posterior probability of the class membership. Fuzzy c-Means by the entropy regularization [5] is regularized hard c-means to obtain a smoothing membership. The objective function is

$$J(U, V, \alpha) = \sum_{k=1}^{n} \sum_{i=1}^{c} u_{ik} D_{KL}(x_k || v_i) + \lambda^{-1} \sum_{k=1}^{n} \sum_{i=1}^{c} u_{ik} \log \frac{u_{ik}}{\alpha_i}. \tag{14}$$

The second term of the right-hand side of (14) is the regularization term. $\alpha = (\alpha_1, \ldots, \alpha_c)$ is a cluster volume size variable with the constraint

$$A = \{\, \alpha = (\alpha_1, \ldots, \alpha_c) \,:\, \sum_{i=1}^{c} \alpha_i = 1,\ \alpha_j \geq 0,\ \forall j \,\}. \tag{15}$$

The relax constraint is assumed for U,

$$M_f = \{\, U = (u_{ik}) \,:\, \sum_{i=1}^{c} u_{ik} = 1,\ u_{jk} \geq 0,\ \forall j, k \,\}. \tag{16}$$

The solution for U is

$$u_{ik} = \frac{\alpha_i \exp(-\lambda D_{KL}(x_k || v_i))}{\sum_{j=1}^{c} \alpha_j \exp(-\lambda D_{KL}(x_k || v_j))}. \tag{17}$$

The solution for V is the same as (22). The solution for α is

$$\alpha_i = \frac{1}{n} \sum_{k=1}^{n} u_{ik}. \tag{18}$$

Algorithm FCM. (Fuzzy c-means)
FCM0. Set an initial value $\bar{V}, \bar{\alpha}$ for V, α.

FCM1. Find optimal solution of J with respect to U while other variables are fixed: put

$$\bar{U} = \arg\min_{U \in M_f} J(U, \bar{V}, \bar{\alpha}).$$

FCM2. Find optimal solution of J with respect to V while other variables are fixed: put

$$\bar{V} = \arg\min_{V} J(\bar{U}, V, \bar{\alpha}).$$

FCM3. Find optimal solution of J with respect to α while other variables are fixed: put

$$\bar{\alpha} = \arg\min_{\alpha \in A} J(\bar{U}, \bar{V}, \alpha).$$

FCM4. If the solution $(\bar{U}, \bar{V}, \bar{\alpha})$ is convergent, stop; else go to **FCM1**.
End of FCM.

5 c-Means Clustering Based on the Hellinger Distance

We also apply the Hellinger distance to hard c-means. The Hellinger distance is closely related with the geodesic distance [3]. Now we consider the squared Hellinger distance,

$$D_H^2(x, x') = \sum_{j=1}^{m+1} \left(\sqrt{x_{kj}} - \sqrt{v_{ij}} \right)^2. \tag{19}$$

Therefore, The objective function is

$$J = \sum_{k=1}^{n} \sum_{i=1}^{c} u_{ik} D_H^2(x_k, v_i). \tag{20}$$

The cluster assignment is almost the same as (12).

$$l = \arg\min_{1 \le i \le c} D_H^2(x_k, v_i) \tag{21}$$

Cluster centers are also derived from the Lagrange multiplier method with the constraint.

$$L = J + \sum_{i=1}^{c} \mu_i \Big(\sum_{j=1}^{m+1} v_{ij} - 1 \Big)$$

$$\frac{\partial L}{\partial v_{ij}} = -\sum_{k=1}^{n} \frac{u_{ik}(\sqrt{x_{kj}} - \sqrt{v_{ij}})}{\sqrt{v_{ij}}} + \mu_i = 0$$

As a result, cluster centers are calculated in the different way from a usual hard c-means.

$$v_{ij} = \frac{\sum_{k=1}^{n} \sum_{l=1}^{n} u_{ik} u_{il} x_{kj} x_{lj}}{\sum_{k=1}^{n} \sum_{l=1}^{n} u_{ik} u_{il} \langle x_k, x_j \rangle} \tag{22}$$

6 Illustrative Example

First, we show equal distance contours from one point in $\mathbb{P}^m$. The contour by the Euclidean distance is shown in Fig. 1, The contour by the KL-divergence is shown in Fig. 2, and The contour by the Hellinger distance is shown in Fig. 3. They show the Hellinger distance emphasize the geom try of the manifold. Second, Artificially generated 60 points on $2d$-simplex were analyzed. We applied hard c-means with three different dissimilarities to data. Fig. 4 shows the result from hard c-means with the Euclidean distance, Fig. 5 shows the result from hard c-means with the KL-divergence, and Fig. 6 shows the result from hard c-means with the Hellinger distance. Fig. 6 is quite different from other two clustering results. The Hellinger distance may be the most appropriate dissimilarity on the manifold. Fig. 7, 8, 9 show equal distance contours from cluster centers.

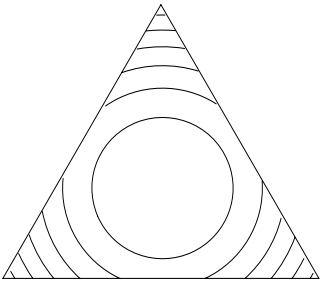


Fig. 1. Equal Euclidean distance contour on $\mathbb{P}^m$

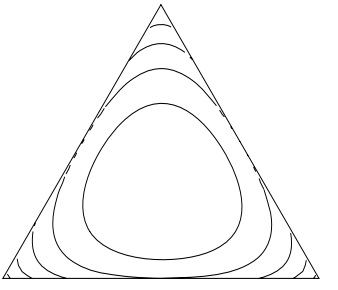


Fig. 2. Equal KL-divergence contour on $\mathbb{P}^m$

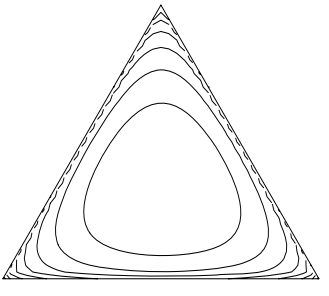


Fig. 3. Equal Hellinger distance contour on $\mathbb{P}^m$

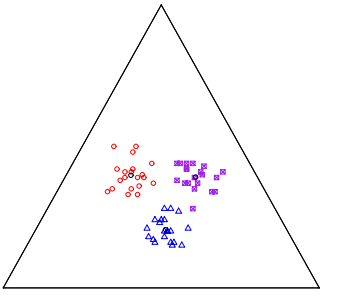


Fig. 4. Result from Hard c-means with the Euclidean distance

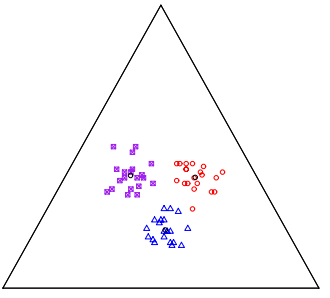


Fig. 5. Result from Hard c-means with the KL-divergence

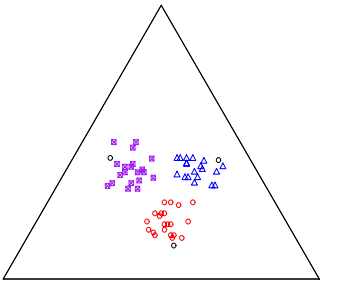


Fig. 6. Result from Hard c-means with the Hellinger distance

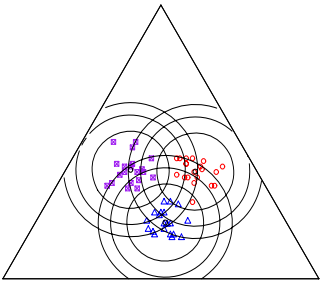


Fig. 7. Equal Euclidean distance contour from cluster centers

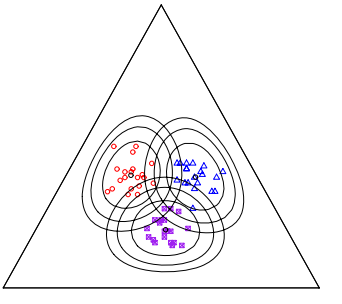


Fig. 8. Equal KL-divergence contour from cluster centers

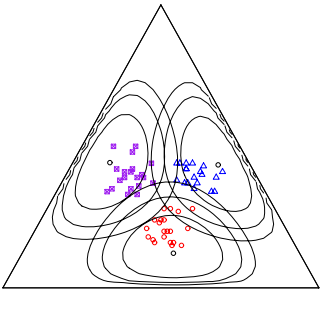


Fig. 9. Equal Hellinger distance contour from cluster centers

7 Conclusion

In this paper, we discuss c-means clustering on the multinomial manifold. Instead of the geodesic distance, the KL-divergence and the Hellinger distance are used in c-means clustering. The Hellinger distance is a good approximation of the geodesic distance compared to the KL-divergence. These metrics are more useful to detect clusters on the manifold than the Euclidean distance.

Future studies include real-world applications such as document clustering, and developments of clustering algorithms using the information geometry.

References

1. Jebara, T., Kondor, R., Howard, A.: Probability product kernels. J. Mach. Learn. Res. 5, 819–844 (2004)
2. Amari, S., Nagaoka, H. (eds.): Methods of Information Geometry. Translations of Mathematical monographs, vol. 191. Oxford University Press, Oxford (2000)
3. Lafferty, J., Lebanon, G.: Diffusion kernels on statistical manifolds. J. Mach. Learn. Res. 6, 129–163 (2005)
4. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York, NY, USA (1981)
5. Miyamoto, S., Mukaidono, M.: Fuzzy c-means as a regularization and maximum entropy approach. In: Proc.of the 7th International Fuzzy Systems Association World Congres (IFSA'97), vol. 2, pp. 86–92 (1997)
6. Dhillon, I.S., Mallela, S., Modha, D.S.: Information-theoretic co-clustering. In: KDD '03. Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 89–98. ACM Press, New York, NY, USA (2003)
7. Pereira, F., Tishby, N., Lee, L.: Distributional clustering of english words. In: Proceedings of the 31st annual meeting on Association for Computational Linguistics, Morristown, NJ, USA, Association for Computational Linguistics, pp. 183–190 (1993)
8. Zhang, D., Chen, X., Lee, W.S.: Text classification with kernels on the multinomial manifold. In: SIGIR '05. Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 266–273. ACM Press, New York, NY, USA (2005)

On a Rough Sets Based Tool for Generating Rules from Data with Categorical and Numerical Values

Hiroshi Sakai[1], Kazuhiro Koba[1], Ryuji Ishibashi[1], and Michinori Nakata[2]

[1] Department of Mathematics and Computer Aided Science, Faculty of Engineering, Kyushu Institute of Technology, Tobata, Kitakyushu 804, Japan
sakai@mns.kyutech.ac.jp
[2] Faculty of Management and Information Science, Josai International University, Gumyo, Togane, Chiba 283, Japan
nakatam@ieee.org

Abstract. Rough set theory has mainly been applied to data with categorical values. In order to handle data with numerical values, we have defined numerical patterns with two symbols # and @, and have proposed more flexible rough sets based rule generation. The concepts of '*coarse*' and '*fine*' for rules are explicitly defined according to numerical patterns. This paper focuses on the rough sets based method for rule generation, which is enhanced by numerical patterns, and refers to the tool programs. Tool programs are applied to data in UCI Machine Learning Repository, and some useful rules are obtained.

Keywords: Rough sets, Rule generation, Numerical values, Tool, Numerical patterns.

1 Introduction

Recently, rough sets based concepts have been applied to several research area [1-7]. We have also proposed a framework *Rough Non-deterministic Information Analysis* (*RNIA*), which handles not only tables with deterministic information but also tables with non-deterministic information [8-10]. However, we implicitly handled tables with categorical values, and we omitted tables with numerical values. Because, multivariate analysis has mainly been employed for tables with numerical values. In order to enhance the framework of *RNIA*, it is necessary to handle tables with numerical values.

In this paper, we focus on rough sets based rule generation with numerical patterns [11]. Generally, rules are defined by implications $Condition \Rightarrow Decision$, which satisfy some appropriate constraints. Then, we refer to the realized tool programs. Finally, we compare previous research with our research.

2 An Example and Rough Sets Based Issues

Let us consider Table 1, which is a part of baseball game data. This is small size data, however it is enough to discuss rough sets based issues.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 269–281, 2007.

Table 1. Players' Batting Data in Baseball Games, AVG: Batting Average, SF&SH: Sacrifice Flies and Hits, SB: Stolen Bases, OBP: On-Base Percentage, SLG: Slugging Percentage

$OBJECT(Players)$	AVG	$SF\&SH$	SB	OBP	SLG
p_1	0.322	0	03	0.397	0.553
p_2	0.312	1	07	0.391	0.430
p_3	0.309	0	03	0.390	0.557
p_4	0.300	0	01	0.307	0.556
p_5	0.273	0	05	0.326	0.467
p_6	0.402	0	02	0.362	0.628
p_7	0.274	3	11	0.327	0.437
p_8	0.271	1	03	0.361	0.466
p_9	0.266	0	00	0.292	0.525
p_{10}	0.263	0	03	0.294	0.363

Every value in AVG is in the form of 0.??? (?: 0, 1, $\cdots$, 9). We name such formulas, like 0.???, *types* of values. In Table 1, the values 03 and 07 in SB implies 3 or 7, respectively. In order to handle numerical values in the same form, we may add some 0 digits to every value.

Now, let us survey rough sets based concepts [1,2] according to Table 1. For an attribute ATR, let $eq(ATR)$ denote a set of all equivalence classes for ATR, and let $[p_i]_{ATR}$ denote an equivalence class with player p_i. In Table 1, let us consider such a relation that two players p_i and p_j are related if attribute values are the same. According to this relation, we have

$$eq(AVG)=\{\{p_1\},\{p_2\},\{p_3\},\{p_4\},\{p_5\},\{p_6\},\{p_7\},\{p_8\},\{p_9\},\{p_{10}\}\},$$
$$eq(SLG)=\{\{p_1\},\{p_2\},\{p_3\},\{p_4\},\{p_5\},\{p_6\},\{p_7\},\{p_8\},\{p_9\},\{p_{10}\}\}.$$

Since $[p_i]_{AVG} \subseteq [p_i]_{SLG}$ holds for every p_i, there exists data dependency from AVG to SLG [1,2]. The degree of dependency is 1.0, and it is possible to obtain consistent implications from Table 1. We see such consistent implications as candidates of rules. Intuitively, 'consistent' means that the same condition concludes the same decision. For example, an implication $[SB, 07] \Rightarrow [AVG, 0.312]$ from p_2 is consistent, and an implication $[SB, 03] \Rightarrow [AVG, 0.322]$ from p_1 is not consistent, because there is an implication $[SB, 03] \Rightarrow [AVG, 0.309]$ from p_3. Here, every $[p_i] \in eq(AVG)$ is a singleton set $\{p_i\}$. Therefore, every degree of dependency from AVG to any attribute is always 1.0. Although we can obtain consistent implications like $[AVG, val] \Rightarrow Decision$, every implication represents just a player's property, and such a consistent implication does not represent the total players' property.

In rough set theory, we usually handle a finite set of categorical values, and implicitly the number of attribute values is restricted to small size. Therefore, we have small number of equivalence classes for every attribute. In an attribute AVG in Table 1, attribute values are decimal numbers between 0.000 and 1.000. The amount of attributes values is 1001. This amount seems too large in rough set theory. For such reason, it is necessary to reduce the number of all equivalence classes.

3 Numerical Patterns, Coarse and Fine Concepts

This section defines numerical patterns, coarse and fine concepts.

3.1 Meaningful Figures and Numerical Patterns

Now in this section, we propose a concept of meaningful figures in numerical values. Let us consider an irrational number π=3.14$\cdots$. For students in elementary schools, π=3.14 will be sufficient for calculating the area of a circle. On the other hand, π=3.14 may be insufficient for researchers of numerical analysis.

Here, we introduce two symbols @ and #, which represent numeric from 0 to 9. We implicitly employ 10-adic numbers for expressing numerical data. A numerical pattern is a sequence of @ and #, for example @@@, @@#, @##, @@.@ and @#.#. Here, '.' denotes a decimal point, and @ does not occur after #. We see @@@, @@#, @## and ### have the same type ???. Three patterns @@.@, @@.# @#.# have the same type ??.?, too. Here, @ denotes a significant figure and # denotes a figure, which we do not care. For example, AVG values of p_3 and p_4 are 0.309 and 0.300, respectively. These two values are different according to a numerical pattern 0.@@@, but these two values are the same according to a numerical pattern 0.@@#. By means of introducing such numerical patterns, we explicitly define the meaningful figures in numerical values.

3.2 Precision, Fine and Coarse Equivalence Classes

Numerical patterns cause different kinds of equivalence classes for the same data. For an attribute ATR and a numerical pattern $patt$, let $eq(ATR, patt)$ denote a set of all equivalence classes according to $patt$. In Table 1, we have the following:

$\mathtt{eq(AVG,0.@\#\#)}=\{\{p_1,p_2,p_3,p_4\},\{p_5,p_7,p_8,p_9,p_{10}\},\{p_6\}\}$.
$\mathtt{eq(AVG,0.@@\#)}=\{\{p_1\},\{p_2\},\{p_3,p_4\},\{p_5,p_7,p_8\},\{p_6\},\{p_9,p_{10}\}\}$.
$\mathtt{eq(AVG,0.@@@)}=\{\{p_1\},\{p_2\},\{p_3\},\{p_4\},\{p_5\},\{p_6\},\{p_7\},\{p_8\},\{p_9\},\{p_{10}\}\}$.
$\mathtt{eq(SF\&SH,@)}=\{\{p_1,p_3,p_4,p_5,p_6,p_9,p_{10}\},\{p_2,p_8\},\{p_7\}\}$.
$\mathtt{eq(SB,@\#)}=\{\{p_1,p_2,p_3,p_4,p_5,p_6,p_8,p_9,p_{10}\},\{p_7\}\}$.
$\vdots \quad \vdots \quad \vdots$
$\mathtt{eq(SLG,0.@\#\#)}=\{\{p_1,p_3,p_4,p_9\},\{p_2,p_5,p_7,p_8\},\{p_6\},\{p_{10}\}\}$.
$\mathtt{eq(SLG,0.@@\#)}=\{\{p_1,p_3,p_4\},\{p_2,p_7\},\{p_5,p_8\},\{p_6\},\{p_9\},\{p_{10}\}\}$.
$\mathtt{eq(SLG,0.@@@)}=\{\{p_1\},\{p_2\},\{p_3\},\{p_4\},\{p_5\},\{p_6\},\{p_7\},\{p_8\},\{p_9\},\{p_{10}\}\}$.

Clearly, it is possible to introduce an order ($<_p$) on precision into a set of patterns with the same type. The order is

$\mathtt{0.@@@} <_p \mathtt{0.@@\#} <_p \mathtt{0.@\#\#} <_p \mathtt{0.\#\#\#}$

for a type 0.???. The numerical pattern 0.@@@ is the most precise, and the numerical pattern 0.### is the most coarse.

Proposition 1. For $eq(ATR)$ and $eq(ATR')$, we define that $eq(ATR) \subseteq eq(ATR')$ holds, if $[x]_{ATR} \subseteq [x]_{ATR'}$ holds for every object x. Let $PATT$ denote a set of numerical patterns (on a set ATR of attributes) with the same type. For $patt_1, patt_2 \in PATT$, let us suppose $patt_1 <_p patt_2$. Then, $eq(ATR, patt_1) \subseteq eq(ATR, patt_2)$ holds.

The concept of precision has already been introduced into rough set theory, and $eq(ATR, patt_1)$ is called *more fine* and $eq(ATR, patt_2)$ is called *more coarse* in [1]. Recently such fine and coarse information is picked up [12,13,14]. Equivalence relations $eq(ATR, patt)$, which depends upon numerical pattern *patt*, give a real instance of the framework defined in [12,13,14].

Definition 1. Let $[x]_{ATR,patt}$ denote an equivalence class with numerical patterns *patt*. For $patt_1 <_p patt_2$, we say $[x]_{ATR,patt_2}$ is more *coarse* than $[x]_{ATR,patt_1}$ and $[x]_{ATR,patt_1}$ is more *fine* than $[x]_{ATR,patt_2}$.

Definition 2. For $[x]_{ATR,patt}$, we replace every @ symbol in *patt* with the attribute value of x for ATR. Then, we have a value, which may contain # symbol. Let $val_{ATR,patt}$ denote this value. We name $[ATR, val_{ATR,patt}]$ a *descriptor* with *numerical pattern patt*. For $patt_1 <_p patt_2$, we say $[ATR, val_{ATR,patt_1}]$ is more *fine* than $[ATR, val_{ATR,patt_2}]$ and $[ATR, val_{ATR,patt_2}]$ is more *coarse* than $[ATR, val_{ATR,patt_1}]$.

Example 1. For numerical patterns 0.@## and 0.@@# in attribute AVG, the order $0.@@\# <_p 0.@\#\#$ holds, and $[p_1]_{AVG,0.32\#} \subseteq [p_1]_{AVG,0.3\#\#}$ also holds. $[p_1]_{AVG,0.32\#}$ is more fine than $[p_1]_{AVG,0.3\#\#}$. In an equivalence class $[p_1]_{AVG,0.3\#\#}$, [AVG,0.3##] is the descriptor with numerical pattern 0.@##. [AVG,0.32#] is more fine than [AVG,0.3##].

4 A Merit of Employing Numerical Patterns

4.1 A Real Execution Handling Standard Descriptors

It is possible to apply programs in [8,10] to Table 1. The following is a real execution for generating *minimal consistent rules* [10] in the form of $Conditions \Rightarrow [SLG, 0.556]$.

```
?-minimal.
<<Minimal Certain Rules from object 4>>
  [1,0.300]=>[5,0.556] [1/1(=1/1,1/1),DGC],
    This rule covers objects [p4],[(0.1,0.1),(1.0,1.0)]
  [3,1]=>[5,0.556] [1/1(=1/1,1/1),DGC],
    This rule covers objects [p4],[(0.1,0.1),(1.0,1.0)]
  [4,0.307]=>[5,0.556] [1/1(=1/1,1/1),DGC],
    This rule covers objects [p4],[(0.1,0.1),(1.0,1.0)]
EXEC_TIME=0.008(sec)
yes
?-disfunc(4,M).
M=[[1,[1,0.300],[3,1],[4,0.307]],[2,[1,0.300],[2,0],[3,1],[4,0.307]],
  [3,[1,0.300],[3,1],[4,0.307]],[5,[1,0.300],[3,1],[4,0.307]],
  [6,[1,0.300],[3,1],[4,0.307]],[7,[1,0.300],[2,0],[3,1],[4,0.307]],
  [8,[1,0.300],[2,0],[3,1],[4,0.307]],[9,[1,0.300],[3,1],[4,0.307]],
  [10,[1,0.300],[3,1],[4,0.307]]]
yes
```

Every attribute is displayed as its ordinal number, for example, SLG is displayed as whose ordinal number 5. The decision part $[SLG, 0.556]$ is so strict that every implication represents only p_4. For every implication τ ($= Condition \Rightarrow Decision$), each list [(0.1,0.1),(1.0, 1.0)] implies the minimum and the maximum values of $support(\tau)$ and $accuracy(\tau)$, which are the most familiar criteria for rule generation.

$support(\tau)$=$number\_of\_the\_implication\_\tau$ / $number\_of\_all\_players$,
$accuracy(\tau)$=$number\_of\_the\_implication\_\tau$ / $number\_of\_the\_condition$.

In non-deterministic information, the minimum and the maximum values may be different. Program *minimal* was implemented for handling tables with non-deterministic information. Of course, program *minimal* can handle tables with deterministic information. Program $disfunc(4, M)$ generates a discernibility function [10,15], which is in the form of disjunctive normal form. For example, the first term [[1,[1,0.300],[3,1],[4,0.307]] in M implies that p_1 can be discriminated from p_4 by using either of the descriptor $[AVG, 0.300]$, $[SB, 1]$ or $[OBP, 0.307]$ from p_4. In program *minimal*, this function M is generated, and three minimal solutions, which assign true to the function M, are obtained. Every solution becomes the condition part of the rule.

4.2 New Candidates of Rules with Numerical Patterns

Program *minimal* handles standard descriptors, and does not handle descriptors with numerical patterns, so it is impossible to obtain such a minimal consistent implication

τ_1`:[OBP,0.39#]`$\Rightarrow$`[AVG,0.3##],`
`support(`τ_1`)=0.3 and accuracy(`τ_1`)=1.0 (consistent),`

which represents three players p_1, p_2 and p_3. The following will be an admissible minimal consistent implication, too.

τ_2`:[OBP,0.3##]`$\wedge$`[SLG,0.5##]`$\Rightarrow$`[AVG,0.3##],`
`support(`τ_2`)=0.3 and accuracy(`τ_2`)=1.0 (consistent).`

Furthermore, we can easily define the precision, i.e., coarse and fine concepts, on rules by means of numerical patterns. For two implications

τ`:[CON,val`$_{\mathtt{CON}}$`]`$\Rightarrow$`[DEC,val`$_{\mathtt{DEC}}$`],` τ'`:[CON,val`$'_{\mathtt{CON}}$`]`$\Rightarrow$`[DEC,val`$'_{\mathtt{DEC}}$`]`
`[CON,val`$'_{\mathtt{CON}}$`]`$<_p$`[CON,val`$_{\mathtt{CON}}$`] or [DEC,val`$'_{\mathtt{DEC}}$`]`$<_p$`[DEC,val`$_{\mathtt{DEC}}$`],`

τ is more coarse than τ', and τ' is more fine than τ. Generally, $support(\tau) \geq support(\tau')$ holds, and τ covers more objects than τ'. In order to increase $accuracy(\tau)$, we may replace descriptors in the condition part with more fine descriptors. On the other hand, we may replace descriptors in the condition part with more coarse descriptors in order to increase $support(\tau)$. According to such manipulation, we can easily define more coarse implications and more fine implications.

5 Rule Generation with Numerical Patterns

In this section, we employ not only standard descriptors but also descriptors with numerical patterns, and focus on generating implications with such descriptors.

5.1 Two Strategies for Rule Generation

We employ the following two strategies for rule generation.

Support Based Strategy (SB-strategy): For a table, let DEC be decision attributes and let η be a tuple of decision attributes values for DEC. Then, find every implication $\tau : Condition \Rightarrow [DEC, \eta]$ satisfying (1) and (2) below;
(1) $support(\tau) \geq \alpha$ for a threshold value α $(0 < \alpha \leq 1)$,
(2) $accuracy(\tau)$ is maximum.

Consistency Based Strategy (CB-strategy): For a table, let DEC be decision attributes and let η be a tuple of decision attributes values for DEC. Then, find every implication $\tau : Condition \Rightarrow [DEC, \eta]$ satisfying (1) and (2) below;
(1) τ is consistent,
(2) $support(\tau)$ is maximum.

Every strategy has each property. In SB-strategy, the condition $support(\tau) \geq \alpha$ is assured, but $accuracy(\tau)$ may be small. In CB-strategy the consistency of τ is assured, but $support(\tau)$ may be small. For data with consistency, we may employ CB-strategy, and we may employ SB-strategy for data without consistency.

5.2 Rule Generation by Support Based Strategy

In Table 1, if we employ either $eq(AVR, 0.@@@)$, $eq(OBP, 0.@@@)$ or $eq(SLG, 0.@@@)$, $support(\tau)$ is 0.1 for every implication. For α=0.3, there is no implication with attributes AVR, OBP nor SLG. For α=0.3 in Table 1, it is necessary to obtain such an implication $\tau : [CON, val_{CON}] \Rightarrow [DEC, val_{DEC}]$ that τ occurs more than $3(=0.3 \times 10)$ times. In this case, both $[CON, val_{CON}]$ and $[DEC, val_{DEC}]$ must occur more than 3 times at least. Such a property is employed in Apriori Algorithm [16,17]. We follow this Apriori algorithm, and revise this algorithm for handling numerical patterns.

Algorithm 1. (An Overview of Support Based Rule Generation)
(SB-1) Fix a threshold α and a decision attribute DEC.
(SB-2) Let OB denote the total set of objects. Find candidates of descriptors CAN_1 with numerical patterns,
CAN_1=$\{[A, val_{A,patt}] | |[x]_{A,val_{A,patt}}| \geq \alpha \times |OB|$ holds for an object $x\}$.
(SB-3) Generate CAN_2;
CAN_2=$\{[A, val_{A,patt1}] \wedge [DEC, val_{DEC,patt2}] |$
$|[x]_{\{A,DEC\},\{val_{A,patt1}, val_{DEC,patt2}\}}| \geq \alpha \times |OB|$ holds for an object x,
$[A, val_{A,patt1}] \in CAN_1, A \neq DEC\}$,
and examine $accuracy(\tau)$ in every implication τ
$\tau : [A, val_{A,patt1}] \Rightarrow [DEC, val_{DEC,patt2}]$.
If $accuracy(\tau)$=1.0, this implication satisfies the condition. We store this implication, and remove $[A, val_{A,patt1}] \wedge [DEC, val_{DEC,patt2}]$
from CAN_2. We also remove the conjunction, which implies a redundant implication of previously obtained.
(SB-4) Generate CAN_n recursively and obtain rules until CAN_n=$\{\}$.

In Algorithm 1, we may not fix decision attribute values in SB-strategy. Let us apply Algorithm 1 to Table 1, and simulate Algorithm 1.

Example 2. Let us fix a threshold α=0.3, and DEC=$\{AVG\}$. Since $0.3 \times |OB|$=0.3×10=3, we consider descriptors which occur more than 3 times in Table 1. In (SB-2), we obtain CAN_1 in the following.

```
CAN1={[1,0.3##],[1,0.2##],[1,0.27#],[2,0],[3,0#],[3,3],
[4,0.3##],[4,0.39#],[5,0.5##],[5,0.4##],[5,0.55#]}.
```

Every descriptor occurs more than 3 times in Table 1. Every implication consists of some descriptors, so any implication with $[A, val_A] \notin CAN_1$ occurs less than 2 times in Table 1, and such an implication τ does not satisfy $support(\tau) \geq 0.3$. It is enough to consider descriptors just in CAN_1. In (SB-3), we have

```
CAN2={[1,0.3##][2,0],[1,0.3##][3,0#],[1,0.3##][4,0.3##],
[1,0.3##][4,0.39#],[1,0.3##][5,0.5##],[1,0.3##][5,0.55#],
[1,0.2##][2,0],[1,0.2##][3,0#],[1,0.2##][4,0.3##],
[1,0.2##][5,0.4##],[1,0.27#][4,0.3##],[1,0.27#][5,0.4##]}.
```

Every element in CAN_2 is a conjunction of $[A, val_A], [1, val_1] \in CAN_1$ $(A \neq 1)$, and for example, we see the first conjunction $[1, 0.3\#\#][2, 0]$ as an implication $[SF\&SH, 0] \Rightarrow [AVG, 0.3\#\#]$. In CAN_2, we calculate $accuracy(\tau)$ of 12 conjunctions respectively, and we have the following two implications whose $accuracy()$ is 1.0,

```
τ1: [4,0.39#]=>[1,0.3##],
τ3: [5,0.55#]=>[1,0.3##].
```

Since $accuracy(\tau)$=1.0 means $accuracy$ of τ is maximum, these two implications are rules to be obtained. The first implication τ_1 has already been shown in section 4.2. We remove the above two conjunctions from CAN_2, and repeat this procedure until CAN_n=$\{\}$. In (SB-4), we obtain CAN_3, and we have an implication in the following

```
τ2: [4,0.3##]∧[5,0.5##]=>[1,0.3##],
```

whose $accuracy()$ is 1.0. This implication τ_2 has also been shown in section 4.2. In the next step, we have CAN_4 in the following

```
CAN4={[1,0.3##][2,0][3,0#][4,0.3##],
[1,0.3##][2,0][3,0#][5,0.5##]}.
```

The both $accuracy()$ values are less than 1.0 in CAN_4, and CAN_5 becomes an empty set $\{\}$. According to these procedures, we obtain three rules whose $support()$ are more than 0.3 and $accuracy()$ are 1.0, respectively.

5.3 Computational Issues on Support Based Strategy

Let OB, AT and $PATT_A$ be a set of all objects, a set of all attributes, and a set of all patterns in $A \in AT$, respectively. In (SB-2), we first prepare arrays for handling the occurrence of $[A, val_{A,patt}]$ $(A \in AT, patt \in PATT_A)$. By means of examining every attribute value in x and A, it is possible to obtain the occurrence of $[A, val_{A,patt}]$. This complexity depends upon $|OB| \times |AT|$. At the same time, we

Table 2. Numerical Patterns for discriminating p_5 to p_{10} from p_1

Player	*SF&SH*	*SB*	*OBP*	*SLG*
p_5		03	0.39#	0.5##
p_6		03	0.39#	0.5##
p_7	0	0#	0.39#	0.5##
p_8	0		0.39#	0.5##
p_9		03	0.3##	0.55#
p_{10}			0.3##	0.5##

store a set of objects satisfying $[A, val_{A,patt}]$. For every descriptor $[A, val_{A,patt}]$, we examine the condition that the occurrence is more than $\alpha \times |OB|$, and we obtain CAN_1.

In (SB-3), we generate conjunctions of descriptors. Let num_{CON} be the number of descriptors $[A, val]$ $(A \neq DEC)$ in CAN_1, and let num_{DEC} be the number of descriptors $[DEC, val]$ in CAN_1. Then, the number of conjunctions is $num_{CON} \times num_{DEC}$. For every conjunction $[A, val_{A,patt1}] \wedge [DEC, val_{DEC,patt2}]$ $(A \neq DEC)$, it is possible to obtain the occurrence by means of applying the intersection operation of corresponding two sets.

We repeat the above procedure. The most time-consuming part is to generate CAN_i (i=2,3,$\cdots$), especially CAN_2 and CAN_3. This part depends upon $num_{CON} \times num_{DEC}$.

5.4 Rule Generation by Consistency Based Strategy

In CB-strategy, we follow the method by discernibility functions [15], which we have shown in section 4.1, and revise this method for handling numerical patterns. The following is the overview of this algorithm.

Algorithm 2. (An Overview of Consistency Based Rule Generation)
(CB-1) Fix a decision attribute DEC and decision attribute values η.
(CB-2) Obtain an equivalence class CL which is defined by a descriptor $[DEC, \eta]$.
(CB-3) For an object $x \in CL$ and a set OB of all objects, generate a discernibility function below;
$DF(x)$=$\wedge_{y \in OB-CL} Disc(x, y)$, $Disc(x, y)$ is a disjunction of descriptors, which discriminate y from x.
(CB-4) The condition part is obtained as a solution of the discernibility function.
According to Table 1, we simulate Algorithm 2.

Example 3. Let us fix DEC=$\{AVG\}$ and η=0.3##, which means to handle implications like $Condition \Rightarrow [AVG, 0.3\#\#]$. In (CB-2), this descriptor $[AVG, 0.3\#\#]$ defines CL=$\{p_1, p_2, p_3, p_4\}$. In (CB-3), we pick up p_1, and we consider conditions which discriminate p_5 to p_{10} from p_1. The most coarse attribute values are in Table 2. For example, $[SLG, 0.5\#\#]$ is enough for discriminating p_5 ($[SLG, 0.467]$) from p_1 ($[SLG, 0.553]$). Of course, both $[SLG, 0.55\#]$ and $[SLG, 0.553]$ can discriminate p_5 from p_1. According to Table 2, we obtain a discernibility function

$DF(p_1) = Disc(p_1, p_5) \wedge Disc(p_1, p_6) \wedge \cdots \wedge Disc(p_1, p_{10})$
$= ([SB, 03] \vee [OBP, 0.39\#] \vee [SLG, 0.5\#\#])$
$\wedge ([SF\&SH, 0] \vee [SB, 0\#] \vee [OBP, 0.39\#] \vee [SLG, 0.5\#\#])$
$\wedge ([SF\&SH, 0] \vee [OBP, 0.39\#] \vee [SLG, 0.5\#\#])$
$\wedge ([SB, 03] \vee [OBP, 0.3\#\#] \vee [SLG, 0.55\#])$
$\wedge ([OBP, 0.3\#\#] \vee [SLG, 0.5\#\#])$.
Since the condition of descriptor $[OBP, 0.39\#]$ automatically satisfies the condition of $[OBP, 0.3\#\#]$, the descriptor $[OBP, 0.39\#]$ is a solution of $DF(p_1)$. Every p_i (i=5, $\cdots$, 10) clearly does not satisfy this condition $[OBP, 0.39\#]$. Therefore, this condition specifies a subset of CL, namely an implication,

τ_1: [OBP,0.39#]⇒[AVG,0.3##]

is consistent and $support(\tau_1)$=0.3. This is the same implication in section 4.2 and 5.2 by accident. We also obtain the following consistent implications including τ_2 and τ_3.

τ_2: [OBP,0.3##]∧[SLG,0.5##]⇒[AVG,0.3##] ($support(\tau_2)$=0.3),
τ_3: [SLG,0.55#]⇒[AVG,0.3##] ($support(\tau_3)$=0.3),
τ_4: [SF&SH,0]∧[SB,03]∧[OBP,0.3##]⇒[AVG,0.3##] ($support(\tau_4)$=0.2).

Since $support()$ is maximum, we finally obtain τ_1, τ_2 and τ_3 as the result of CB-strategy.

6 An Implementation of a Tool for SB-Strategy and Its Application

We first show the application of this tool, then refer to the implementation.

6.1 An Application

Now, we refer to the application of this tool to data hepatitis.dat in UCI Repository [18]. This data consists of 155 objects and 20 attributes. Here, we omitted 75 objects with missing values, namely data consists of 80 objects and 20 attributes. The following is a part of attributes and attribute values.

```
1.Class:die,live, 2.AGE:numerical, 3.SEX:male,female,
4.STEROID:no,yes, 5.ANTIVIRALS:no,yes, 6.FATIGUE:no,yes,
       :           :           :
18.ALBUMIN:numerical, 19.PROTIME:numerical, 20.HISTOLOGY:no,yes.
```

The following is a part of the real data.

```
80
20
2 34 1 2 2 2 2 2 2 2 2 2 2 2 90 95 28 40 75 1
2 39 1 1 1 2 2 2 1 1 2 2 2 2 130 78 30 44 85 1
2 32 1 2 1 1 2 2 2 1 2 1 2 2 100 59 249 37 54 1
2 41 1 2 1 1 2 2 2 1 2 2 2 2 90 81 60 39 52 1
       :           :           :
```

In this data, all categorical values, like die and live, male and female, no and yes, are replaced with numerical values 1 and 2 for convenience. The first values 80 and 20 implies the number of objects and attributes. In attribute 19.PROTIME, attributes values seem to be between 0 and 100. Therefore in attribute 19.PROTIME, there may be 101 equivalence classes according to the usual rough sets based definition. If we employ such equivalence classes, we may not obtain any implication τ, whose $support(\tau)$ value is high. In such cases, we employ numerical patterns.

For this data, we fix a decision to $[Class, live]$ and $support(\tau) \geq 0.5$, and generated implications $Condition \Rightarrow [Class, live]$. For execution of the program, it is enough to specify data file, a decision and a threshold value, like 0.5. In reality, we obtained an implication

τ_5:[14,2]∧[18,4.#]=>[1,2] [0.500,1.000]

in CAN_3. The values 0.500 and 1.000 implies $support(\tau_5)$ and $accuracy(\tau_5)$, respectively. In every CAN_i $(i \geq 4)$, no implication satisfied the condition that $accuracy()$=1.0. Therefore, τ_5 is the unique implication based on SB-strategy. The attribute values in 18:ALBUMIN are 4.0 to 4.9 except 4.8, namely 9 attributes values. This τ_5 combines these 9 attribute values, and realizes the condition $support(\tau_5) \geq 0.5$. If we do not employ # symbols, τ_5 is missed.

In order to obtain more implications, we loosened such a condition that $accuracy(\tau)$ is maximum to the condition $accuracy(\tau) \geq 0.9$, and we obtained the following.

[18,4.#]=>[1,2] [0.525,0.976],
[13,2]∧[16,0##]=>[1,2] [0.512,0.931],
[13,2]∧[17,0##]=>[1,2] [0.575,0.901],
[13,2]∧[18,4.#]=>[1,2] [0.512,0.976],
[14,2]∧[16,0##]=>[1,2] [0.525,0.913],
[11,2]∧[13,2]∧[17,0##]=>[1,2] [0.500,0.909],
[11,2]∧[13,2]∧[19,0##]=>[1,2] [0.537,0.914],
[12,2]∧[14,2]∧[17,0##]=>[1,2] [0.500,0.930],
[12,2]∧[14,2]∧[19,0##]=>[1,2] [0.512,0.931],
[13,2]∧[14,2]∧[16,0##]=>[1,2] [0.500,0.952],
[13,2]∧[14,2]∧[17,0##]=>[1,2] [0.550,0.916],
[13,2]∧[14,2]∧[19,0##]=>[1,2] [0.587,0.903],
[11,2]∧[13,2]∧[14,2]∧[19,0##]=>[1,2] [0.512,0.931].

In an attribute 16:ALKPHOSPHATE, the descriptor [16,0##] implies the attribute values are less than 99. Every object, whose attribute value is more than 100, does not satisfies the condition of [16,0##]. This holds for attributes 17:SGOT and 19:PROTIME, too. Every implication contains # symbols, and every implication is missed without # symbol. For this execution, it took about 5.770(sec) in a PC with Pentium 3 (747MHz).

6.2 An Implementation of a Tool Program

A tool program *apri_num*() consists of some functions, and simulates Algorithm 1. Every user just specifies data file, a decision and a threshold value. According

to the threshold value, CAN_1 in (SB-2) is automatically generated, and CAN_2, CAN_3, $\cdots$ are sequentially generated. At the same time, implications are obtained as rules.

In (SB-2), function $read\_file()$, which includes data file, is employed at first, and descriptors with numerical patterns are generated by using functions $make\_eq()$ and $change()$. Then, two functions $filter()$ and $apri\_gen()$ are employed repeatedly. Function $filter()$ examines the condition $support(\tau) \geq \alpha$ for τ, and $apri\_gen()$ generates conjunctions of descriptors. The realized program $apri\_num()$ consists of about 500 lines in C.

7 Comparison of Previous Research and Our Research

There will be a large number of research on the discretization of numerical values, like interval theory and cluster analysis. As for rough sets based rule generation, the discretization is investigated [19,20], too. In [19,20], such an issue that how we divide an interval into subintervals is mainly considered, and the entropy function based discretization is proposed. This method is also employed in the decision tree generation, like C4.5. Rules are generated from such discretized data.

In our research, the purpose is to handle the coarse and fine concepts, which are defined by descriptors with numerical patterns, in rule generation. For example, $[AVG, 0.3\#\#]$ in Table 1 implies that this player's batting average is 30 percent. Such information is familiar in our life. In order to handle such coarse information explicitly in rough set theory, we introduced numerical patterns into rough sets based rule generation. The purpose of our research is not to discretize numerical values, but every numerical pattern defines a discretization over attribute values as a result. Namely,

- In the previous research, the discretization of numerical values is proposed, and the rough sets based method is employed. Therefore, obtained rules will be characterized by the discretization.
- In our research, we extended the rough sets based method to a method with numerical patterns. According to numerical patterns, the discretization is naturally defined, and the coarse and fine concepts are defined. Therefore, obtained rules will be characterized by descriptors with numerical patterns.

In both previous research and our research, the discretization of numerical values is necessary, however previous research handles rule generation based on the property of data, and our research handles rule generation based on the possible expression of descriptors. In this point, these research will be different kinds of research.

8 Concluding Remarks

In order to handle data with numerical values, we have proposed numerical patterns. According to numerical patterns, it is possible to define an order of

precision on patterns with the same type, and coarse and fine concepts over equivalence relations, equivalence classes and descriptors are defined. These fine and coarse equivalence classes cause more flexible rule generation in numerical data. As for tool programs, we have shown the overview of a tool for SB-strategy. We are now realizing tool programs for CB-strategy not in prolog but in C.

Acknowledgment. This work is partly supported by the Grant-in-Aid for Scientific Research (C) (No.18500214), Japan Society for the Promotion of Science.

References

1. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1991)
2. Pawlak, Z.: Some Issues on Rough Sets, Transactions on Rough Sets. In: Peters, J.F., Skowron, A., Grzymała-Busse, J.W., Kostek, B., Świniarski, R.W., Szczuka, M. (eds.) Transactions on Rough Sets I. LNCS, vol. 3100, pp. 1–58. Springer, Heidelberg (2004)
3. Komorowski, J., Pawlak, Z., Polkowski, L., Skowron, A.: Rough Sets: a tutorial, Rough Fuzzy Hybridization, pp. 3–98. Springer, Heidelberg (1999)
4. Nakamura, A., Tsumoto, S., Tanaka, H., Kobayashi, S.: Rough Set Theory and Its Applications. Journal of Japanese Society for AI 11(2), 209–215 (1996)
5. Polkowski, L., Skowron, A. (eds.): Rough Sets in Knowledge Discovery 1, Studies in Fuzziness and Soft Computing, vol. 18. Physica-Verlag, Heidelberg (1998)
6. Polkowski, L., Skowron, A. (eds.): Rough Sets in Knowledge Discovery 2, Studies in Fuzziness and Soft Computing, vol. 19. Physica-Verlag, Heidelberg (1998)
7. Rough Set Software, Bulletin of Int'l. Rough Set Society, vol. 2, pp. 15–46 (1998)
8. Sakai, H., Okuma, A.: Basic Algorithms and Tools for Rough Non-deterministic Information Analysis. In: Peters, J.F., Skowron, A., Grzymała-Busse, J.W., Kostek, B., Świniarski, R.W., Szczuka, M. (eds.) Transactions on Rough Sets I. LNCS, vol. 3100, pp. 209–231. Springer, Heidelberg (2004)
9. Sakai, H.: Effective Procedures for Handling Possible Equivalence Relations in Non-deterministic Information Systems. Fundamenta Informaticae 48, 343–362 (2001)
10. Sakai, H., Nakata, M.: Discernibility Functions and Minimal Rules in Non-deterministic Information Systems. In: Ślęzak, D., Wang, G., Szczuka, M., Düntsch, I., Yao, Y. (eds.) RSFDGrC 2005. LNCS (LNAI), vol. 3641, pp. 254–264. Springer, Heidelberg (2005)
11. Sakai, H., Murai, T., Nakata, M.: On a Tool for Rough Non-deterministic Information Analysis and Its Perspective for Handling Numerical Data. In: Torra, V., Narukawa, Y., Miyamoto, S. (eds.) MDAI 2005. LNCS (LNAI), vol. 3558, pp. 203–214. Springer, Heidelberg (2005)
12. Murai, T., Resconi, G., Nakata, M., Sato, Y.: Operations of Zooming In and Out on Possible Worlds for Semantic Fields. In: Damiani, E., et al. (eds.) Knowledge-Based Intelligent Information Engineering Systems and Allied Technologies, pp. 1083–1087. IOS Press, Amsterdam, Trento, Italy (2002)
13. Murai, T., Resconi, G., Nakata, M., Sato, Y.: Granular Reasoning Using Zooming In & Out. In: Wang, G., Liu, Q., Yao, Y., Skowron, A. (eds.) RSFDGrC 2003. LNCS (LNAI), vol. 2639, pp. 421–424. Springer, Heidelberg (2003)

14. Yao, Y., Liau, C., Zhong, N.: Granular Computing Based on Rough Sets, Quotient Space Theory, and Belief Functions. In: Zhong, N., Raś, Z.W., Tsumoto, S., Suzuki, E. (eds.) ISMIS 2003. LNCS (LNAI), vol. 2871, pp. 152–159. Springer, Heidelberg (2003)
15. Skowron, A., Rauszer, C.: The Discernibility Matrices and Functions in Information Systems. In: Intelligent Decision Support - Handbook of Advances and Applications of the Rough Set Theory, pp. 331–362. Kluwer Academic Publishers, Boston, MA (1992)
16. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules. In: Proc. 20th Very Large Data Base, pp. 487–499 (1994)
17. Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A.: Fast Discovery of Association Rules, Advances in Knowledge Discovery and Data Mining, pp. 307–328 (1996)
18. UCI Machine Repository: http://www.ics.uci.edu/~mlearn/MLRepository.html
19. Chmielewski, M., Grzymala-Busse, J.: Global Discretization of Continuous Attributes as Preprocessing for Machine Learning. Int'l. Journal of Approximate Reasoning 15, 319–331 (1996)
20. Grzymala-Busse, J., Stefanowski, J.: Three Discretization Methods for Rule Induction. Int'l. Journal of Intelligent Systems 16, 29–38 (2001)

Applying Rough Sets to Information Tables Containing Probabilistic Values

Michinori Nakata[1] and Hiroshi Sakai[2]

[1] Faculty of Management and Information Science,
Josai International University
1 Gumyo, Togane, Chiba, 283-8555, Japan
nakatam@ieee.org
[2] Department of Mathematics and Computer Aided Sciences,
Faculty of Engineering, Kyushu Institute of Technology,
Tobata, Kitakyushu, 804-8550, Japan
sakai@mns.kyutech.ac.jp

Abstract. Rough sets are applied to information tables containing imprecise values that are expressed in a probability distribution. A family of weighted equivalence classes is obtained where each equivalence class is accompanied by the probability to which it is an actual one. By using the family of weighted equivalence classes, we derive lower and upper approximations. The lower and upper approximations coincide with ones obtained from methods of possible worlds. Therefore, the method of weighted equivalence classes is justified. In addition, this method is applied to missing values interpreted probabilistically. Using weighted equivalence classes correctly derives a lower approximation, even in the case where the method of Kryszkiewicz does not derive any lower approximation.

Keywords: Rough sets, Imprecise information, Probabilistic value, Weighted equivalence class, Lower and upper approximations.

1 Introduction

Rough sets play a significant role in the field of knowledge discovery and data mining since the first paper published by Pawlak [19]. The framework of rough sets is constructed under the premise that information tables consisting of precise information are obtained. However, there ubiquitously exists imprecise information in the real world [18]. Thus, it has been investigated to apply rough sets to information tables containing imprecise information represented by a missing value, an or-set, a possibility distribution, etc [2,4,5,10,11,13,14,15,20,21,22,23,25]. The methods are broadly separated into three ways.

The first method is one based on possible worlds [17,20,21,22]. In the method, possible tables, which consist of precise values, are obtained from an information table. Each possible table is dealt with by the traditional methods of applying rough sets to information tables containing precise information, and then the results from

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 282–294, 2007.

the possible tables are aggregated. In other words, the methods that are already established are applied to each possible table. Therefore, there is no doubt for correctness of the treatment. However, the method has difficulties for knowledge discovery at the level of a set of possible values, although it is suitable for finding knowledge at the level of possible values. This is because the number of possible tables exponentially increases as the number of imprecise attribute values increases.

The second method is to use assumptions on indiscernibility of missing values [2,5,8,10,11,25]. Under the assumptions, we can obtain a binary relation for indiscernibility between objects. To the binary relation, rough sets are applied by using a class of objects; for example, an indiscernible class. In the method, it is not clarified why the assumptions are valid to real data sets.

The third method directly deals with imprecise values, without using any assumptions for indiscernibility, under extending the traditional method of rough sets [13,14,15,16,25]. In the method, imprecise values are dealt with probabilistically or possibilistically and the traditional methods are probabilistically or possibilistically extended.[1] A binary relation for indiscernibility is constructed by calculating a degree for indiscernibility between objects. Indiscernible classes for each object are obtained from the binary relation for indiscernibility. The correctness criterion is that any extended method has to give the same results as the method of possible worlds [13]. This criterion is commonly used in the field of databases handling imprecise information [1,7,28].

Stefanowski and Tsoukiàs used implication operators to calculate an inclusion degree between indiscernible classes [25]. Nakata and Sakai have shown that the results in terms of implication operators do not satisfy the correctness criterion and has proposed the method that satisfies the correctness criterion [13,14,15]. However, the proposed method has difficulties for definability, because rough approximations are defined by constructing sets from singletons. Therefore, we propose a method using equivalence classes, called a method of weighted equivalence classes. In this paper, we show how weighted equivalence classes are applied to information tables containing imprecise values expressed in a probability distribution, called probabilistic values.[2]

In Section 2, we briefly address the traditional methods of applying rough sets to information tables containing precise information. In Section 3, methods of possible worlds are mentioned. In the methods, the extended set of possible tables is obtained from an information table containing imprecise values. The traditional methods of applying rough sets to precise information deal with each possible table, and then the results from possible tables are aggregated. In Section 4, methods of applying rough sets to information tables containing probabilistic values are described in terms of weighted equivalence classes. In Section 5, the method of weighted equivalence classes is applied to information tables containing missing values under probabilistic interpretation. Section 6 presents conclusions.

[1] Ziarko proposes methods of rough sets applying data tables where each data is accompanied by a probability [26,27].

[2] See the reference [16] for information tables containing possibilistic values expressed in a possibility distribution.

2 Rough Sets to Precise Information

A data set is represented as a table, called an information table, where each row represents an object and each column represents an attribute. The information table is pair $\mathcal{A} = (U, AT)$. U is a non-empty finite set of objects called the universe. Concretely speaking, U is the set of objects that comprise the information table. AT is a non-empty finite set of attributes such that $\forall a \in AT \ : \ U \rightarrow V_a$. Set V_a is called the domain of attribute a. In information table T whose framework is set AT of attributes, binary relation $IND(\Psi_A)$ for indiscernibility of objects in subset $\Psi \subseteq U$ on subset $A \subseteq AT$ of attributes is,

$$IND(\Psi_A) = \{(o, o') \in \Psi \times \Psi \mid \forall a \in A \ \ a(o) = a(o')\}, \tag{1}$$

where $a(o)$ and $a(o')$ denote values of attribute a for objects o and o', respectively. This relation is called an indiscernibility relation. Obviously, $IND(\Psi_A)$ is an equivalence relation. From the indiscernibility relation, equivalence class $E(\Psi_A)_o (= \{o' \mid (o, o') \in IND(\Psi_A)\})$ containing object o is obtained. This is also the set of objects that is indiscernible with object o, called the indiscernible class for object o. Finally, family $U/IND(\Psi_A)$ $(= \{E(\Psi_A)_o \mid o \in \Psi\})$ of equivalence classes is derived from the indiscernibility relation. All equivalence classes obtained from the indiscernibility relation do not intersect with each other. This means that the objects are uniquely partitioned.

Using equivalence classes, lower approximation $\underline{Apr}(\Phi_B, \Psi_A)$ and upper approximation $\overline{Apr}(\Phi_B, \Psi_A)$ of $\Phi/IND(\Phi_B)$ by $\Psi/IND(\Psi_A)$ are,

$$\underline{Apr}(\Phi_B, \Psi_A) = \{E(\Psi_A) \mid \exists E(\Phi_B) \ E(\Psi_A) \subseteq E(\Phi_B)\}, \tag{2}$$

$$\overline{Apr}(\Phi_B, \Psi_A) = \{E(\Psi_A) \mid \exists E(\Phi_B) \ E(\Psi_A) \cap E(\Phi_B) \neq \emptyset\}. \tag{3}$$

where $E(\Psi_A) \in \Psi/IND(\Psi_A)$ and $E(\Phi_B) \in \Phi/IND(\Phi_B)$ are equivalence classes for sets Ψ and Φ of objects on sets A and B of attributes, respectively. These formulas are expressed in terms of a family of equivalence classes. When we express the approximations in terms of a set of objects, the following expressions are used:

$$\underline{apr}(\Phi_B, \Psi_A) = \{o \mid o \in E(\Psi_A) \wedge \exists E(\Phi_B) \ E(\Psi_A) \subseteq E(\Phi_B)\}, \tag{4}$$

$$\overline{apr}(\Phi_B, \Psi_A) = \{o \mid o \in E(\Psi_A) \wedge \exists E(\Phi_B) \ E(\Psi_A) \cap E(\Phi_B) \neq \emptyset\}. \tag{5}$$

3 Methods of Possible Worlds

In methods of possible worlds, the traditional ways addressed in the previous section are applied to each possible table, and then the results from possible tables are aggregated.

When probabilistic values expressed in a probability distribution is contained in information table T, we obtain extended set $rep(T)$ of possible tables,

$$rep(T) = \{(pt_1, \mu(pt_1)), \ldots, (pt_n, \mu(pt_n))\}, \tag{6}$$

where $\mu(pt_i)$ denotes the probability to which possible table pt_i is the actual one and n is equal to $\Pi_{i=1,m} l_i$, where the number of probabilistic values is m and each of them is expressed in a probability distribution having $l_i (i = 1, m))$ elements. When possible table pt_i is the table where probabilistic values in information table T are replaced by v_{i1}, v_{i2}, ..., v_{im},

$$\mu(pt_i) = \prod_{k=1,m} \pi(v_{ik}), \tag{7}$$

where $\prod$ denotes product and probability $\pi(v_{ik})$ of element v_{ik} comes from probability distribution π expressing the probabilistic value to which the element belongs.

Each possible table consists of precise values. Family $U/IND(\Psi_A)_{pt_i}$ of equivalence classes on set A of attributes is obtained from each possible table pt_i. Possible table pt_i is accompanied by probability $\mu(pt_i)$ to which it is the actual information table. Thus, the family of possible equivalence classes accompanied by a probability is obtained for each possible table, which is expressed by $(U/IND(\Psi_A)_{pt_i}, \mu(pt_i))$. When we express $(U/IND(\Psi_A)_{pt_i}, \mu(pt_i))$ in terms of equivalence classes,

$$(U/IND(\Psi_A)_{pt_i}, \mu(pt_i)) = \{(E(\Psi_A), \mu(pt_i)) \mid E(\Psi_A) \in U/IND(\Psi_A)_{pt_i}\}, \tag{8}$$

where equivalence class $E(\Psi_A)$ is a possible equivalence class on set A of attributes and has probability $\mu(pt_i)$ to which it is one of actual equivalence classes. $U/IND(\Psi_A)$ is the union of $(U/IND(\Psi_A)_{pt_i}, \mu(pt_i))$,

$$U/IND(\Psi_A) = \cup_i (U/IND(\Psi_A)_{pt_i}, \mu(pt_i)). \tag{9}$$

Note that the summation of probabilities is taken in the union if there are the same elements accompanied by a probability. When we express family $U/IND(\Psi_A)$ in terms of equivalence classes,

$$U/IND(\Psi_A) = \{(E(\Psi_A), \kappa(E(\Psi_A) \in U/IND(\Psi_A))) \mid \kappa(E(\Psi_A) \in U/IND(\Psi_A)) > 0\}, \tag{10}$$

where probability $\kappa(E(\Psi_A) \in U/IND(\Psi_A))$ to which equivalence class $E(\Psi_A)$ is contained in $U/IND(\Psi_A)$ is,

$$\kappa(E(\Psi_A) \in U/IND(\Psi_A)) = \sum_{E(\Psi_A) \in U/IND(\Psi_A)_{pt_i}} \mu(pt_i). \tag{11}$$

To obtain lower and upper approximations, the traditional methods addressed in the previous section are applied to possible tables. Let $\underline{Apr}(\Phi_B, \Psi_A)_{pt_i}$ and $\overline{Apr}(\Phi_B, \Psi_A)_{pt_i}$ denote the lower and upper approximations of $U/IND(\Phi_B)_{pt_i}$ by $U/IND(\Psi_A)_{pt_i}$ in possible table pt_i having probability $\mu(pt_i)$. $\underline{Apr}(\Phi_B, \Psi_A)_{pt_i}$ and $\overline{Apr}(\Phi_B, \Psi_A)_{pt_i}$ are accompanied by probability $\mu(pt_i)$, which is expressed by $(\underline{Apr}(\Phi_B, \Psi_A)_{pt_i}, \mu(pt_i))$ and $(\overline{Apr}(\Phi_B, \Psi_A)_{pt_i}, \mu(pt_i))$. $\underline{Apr}(\Phi_B, \Psi_A)$ and $\overline{Apr}$

(Φ_B, Ψ_A) are the union of $(\underline{Apr}(\Phi_B, \Psi_A)_{pt_i}, \mu(pt_i))$ and $(\overline{Apr}(\Phi_B, \Psi_A)_{pt_i}, \mu(pt_i))$, respectively.

$$\underline{Apr}(\Phi_B, \Psi_A) = \cup_i(\underline{Apr}(\Phi_B, \Psi_A)_{pt_i}, \mu(pt_i)), \tag{12}$$

$$\overline{Apr}(\Phi_B, \Psi_A) = \cup_i(\overline{Apr}(\Phi_B, \Psi_A)_{pt_i}, \mu(pt_i)). \tag{13}$$

When we express approximations in terms of equivalence classes,

$$\underline{Apr}(\Phi_B, \Psi_A) = \{(E(\Psi_A), \kappa(E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A))) \mid \kappa(E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A)) > 0\}, \tag{14}$$

$$\overline{Apr}(\Phi_B, \Psi_A) = \{(E(\Psi_A), \kappa(E(\Psi_A) \in \overline{Apr}(\Phi_B, \Psi_A))) \mid \kappa(E(\Psi_A) \in \overline{Apr}(\Phi_B, \Psi_A)) > 0\}, \tag{15}$$

where probabilities $\kappa(E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A))$ and $\kappa(E(\Psi_A) \in \overline{Apr}(\Phi_B, \Psi_A))$ to which equivalence class $E(\Psi_A)$ is contained in $\underline{Apr}(\Phi_B, \Psi_A)$ and $\overline{Apr}(\Phi_B, \Psi_A)$ are,

$$\kappa(E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A)) = \sum_{E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A))_{pt_i}} \mu(pt_i), \tag{16}$$

$$\kappa(E(\Psi_A) \in \overline{Apr}(\Phi_B, \Psi_A)) = \sum_{E(\Psi_A) \in \overline{Apr}(\Phi_B, \Psi_A))_{pt_i}} \mu(pt_i). \tag{17}$$

These formulas show that the summation of the probabilities of possible tables where equivalence class $E(\Psi_A)$ is contained in rough approximations is equal to the probability for equivalence class $E(\Psi_A)$.

Proposition 1
When $(E(\Psi_A), \kappa(E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A)))$ is an element of $\underline{Apr}(\Phi_B, \Psi_A)$ in information table T, there exists set $\underline{PT}$ of possible tables where for all $pt \in \underline{PT}$ $\underline{Apr}(\Phi_B, \Psi_A)_{pt}$ contains $E(\Psi_A)$ and $\sum_{pt \in PT} \mu(pt)$ is equal to $\kappa(E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A))$.

Proposition 2
When $(E(\Psi_A), \kappa(E(\Psi_A) \in \overline{Apr}(\Phi_B, \Psi_A)))$ is an element of $\overline{Apr}(\Phi_B, \Psi_A)$ in information table T, there exists set $\overline{PT}$ of possible tables where for all $pt \in \overline{PT}$ $\overline{Apr}(\Phi_B, \Psi_A)_{pt}$ contains $E(\Psi_A)$ and $\sum_{pt \in PT} \mu(pt)$ is equal to $\kappa(E(\Psi_A) \in \overline{Apr}(\Phi_B, \Psi_A))$.

When the lower and upper approximations are expressed in terms of a set of objects,

$$\underline{apr}(\Phi_B, \Psi_A) = \{(o, \kappa(o \in \underline{apr}(\Phi_B, \Psi_A))) \mid \kappa(o \in \underline{apr}(\Phi_B, \Psi_A)) > 0\}, \tag{18}$$

$$\overline{apr}(\Phi_B, \Psi_A) = \{(o, \kappa(o \in \overline{apr}(\Phi_B, \Psi_A))) \mid \kappa(o \in \overline{apr}(\Phi_B, \Psi_A)) > 0\}, \tag{19}$$

and

$$\kappa(o \in \underline{apr}(\Phi_B, \Psi_A)) = \sum_{E(\Psi_A) \ni o} \kappa(E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A)), \tag{20}$$

$$\kappa(o \in \overline{apr}(\Phi_B, \Psi_A)) = \sum_{E(\Psi_A) \ni o} \kappa(E(\Psi_A) \in \overline{Apr}(\Phi_B, \Psi_A)). \tag{21}$$

We adopt results from methods of possible worlds as a correctness criterion of extended methods of applying rough sets to imprecise information. This is commonly used in the field of databases handling imprecise information [1,7,28].

Correctness criterion
Results obtained from applying an extended method to an information table containing imprecise values are the same as ones obtained from applying the corresponding traditional method to every possible table derived from that information table and aggregating the results created in the possible tables.

4 Rough Sets to Information Tables Containing Probabilistic Values

When object o takes imprecise values for attributes, we calculate the degree to which the attribute values are the same as another object o'. The degree is the indiscernibility degree of object o and o' on the attributes. In this case, a binary relation for indiscernibility on set A of attributes is,

$$IND(\Psi_A) = \{((o, o'), \kappa(A(o) = A(o'))) \mid (\kappa(A(o) = A(o')) \neq 0) \wedge (o \neq o')\} \cup \{((o, o), 1)\}, \tag{22}$$

where $\kappa(A(o) = A(o'))$ denotes the indiscernibility degree of objects o and o' on set A of attributes and is equal to $\kappa((o, o') \in IND(\Psi_A))$,

$$\kappa(A(o) = A(o')) = \bigotimes_{a \in A} \kappa(a(o) = a(o')), \tag{23}$$

where operator $\bigotimes$ depends on properties of imprecise attribute values. When the imprecise attribute values are expressed in a probability distribution, the operator is product denoted by $\prod$.

From binary relation $IND(\Psi_A)$ for indiscernibility, family $U/IND(\Psi_A)$ of weighted equivalence classes is obtained via indiscernible sets. Among the elements of $IND(\Psi_A)$, set $S_A(o)$ of objects that are paired with object o, called the indiscernible set on set A of attributes for object o, is,

$$S_A(o) = \{o' \mid \kappa((o, o') \in IND(\Psi_A)) > 0\}. \tag{24}$$

$S_A(o)$ is the greatest possible equivalence class among possible equivalence classes containing objects o, when objects o has a precise value on all attributes in set A

of attributes. Let $PS_A(o)$ denote the power set of $S_A(o)$. From $PS_A(o)$, family $Can(U/IND(\Psi_A)_o)$ of candidates for possible equivalence classes containing object o is obtained,

$$Can(U/IND(\Psi_A)_o) = \{E(\Psi_A) \mid E(\Psi_A) \in PS_A(o) \wedge o \in E(\Psi_A)\}. \quad (25)$$

Whole family $Can(U/IND(\Psi_A))$ of candidates for possible equivalence classes is,

$$Can(U/IND(\Psi_A)) = \cup_o Can(U/IND(\Psi_A)_o). \quad (26)$$

Probability $\kappa(E(\Psi_A) \in U/IND(\Psi_A))$ to which candidate $E(\Psi_A) \in Can(U/IND(\Psi_A))$ is an actual one is,

$$\kappa(E(\Psi_A) \in U/IND(\Psi_A)) = \kappa(\wedge_{o \in E(\Psi_A) \ and \ o' \in E(\Psi_A)}(A(o) = A(o')) \wedge_{o \in E(\Psi_A) \ and \ o' \notin E(\Psi_A)}(A(o) \neq A(o'))), \quad (27)$$

where $o \neq o'$, $\kappa(f)$ is the probability to which formula f is satisfied, and $\kappa(f) = 1$ when there exists no f. When set Ψ of objects contains k objects and equivalence class $E(\Psi_A)$ consists of l objects,

$$\kappa(E(\Psi_A) \in U/IND(\Psi_A)) = \sum_{(u,v_1,\cdots,v_{k-l})} (\prod_{o \in E(\Psi_A)} \pi_{A(o)}(u) \times \prod_{o_i \notin E(\Psi_A)} (\pi_{A(o_1)}(v_1), \pi_{A(o_2)}(v_2), \ldots, \pi_{A(o_{k-l})}(v_{k-l}))), \quad (28)$$

where

$$\pi_{A(o)}(u) = \prod_{j=1,m} \pi_{a_j(o)}(u_j), \quad (29)$$

$$\pi_{A(o_i)}(v_i) = \prod_{j=1,m} \pi_{a_j(o_i)}(v_{ij}), \quad (30)$$

where two values u and v_i are different and are expressed in $(u_1, \cdots, u_m)$ and $(v_{i1}, \cdots, v_{im})$ on set $A(= \{a_1, a_2, \ldots, a_m\})$ of attributes, respectively. Finally, family $U/IND(\Psi_A)$ of weighted equivalence classes is,

$$U/IND(\Psi_A) = \{(E(\Psi_A), \kappa(E(\Psi_A) \in U/IND(\Psi_A))) \mid \kappa(E(\Psi_A) \in U/IND(\Psi_A)) > 0\}. \quad (31)$$

Proposition 3
When $(E(\Psi_A), \kappa(E(\Psi_A) \in U/IND(\Psi_A)))$ is an element of $U/IND(\Psi_A)$ in information table T, there exists set PT of possible tables where for all $pt \in PT$ $U/IND(\Psi_A)_{pt}$ contains $E(\Psi_A)$ and $\sum_{pt \in PT} \mu(pt)$ is equal to $\kappa(E(\Psi_A) \in U/IND(\Psi_A))$.

Proposition 4
$U/IND(\Psi_A)$ in an information table is equal to the union of the families of possible equivalence classes accompanied by a probability, where each family of possible equivalence classes is obtained from a possible table created from the information table.

Proposition 5
For any object o,

$$\sum_{E(\Psi_A)\ni o} \kappa(E(\Psi_A) \in U/IND(\Psi_A)) = 1. \tag{32}$$

Using families of weighted equivalence classes, we can obtain lower approximation $\underline{Apr}(\Phi_B, \Psi_A)$ and upper approximation $\overline{Apr}(\Phi_B, \Psi_A)$ of $U/IND(\Phi_B)$ by $U/IND\overline{(\Psi_A)}$. For the lower approximation,

$$\underline{Apr}(\Phi_B, \Psi_A) = \{(E(\Psi_A), \kappa(E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A))) \mid \kappa(E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A)) > 0\}, \tag{33}$$

$$\kappa(E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A)) = \sum_{E(\Phi_B)} (\kappa(E(\Psi_A) \subseteq E(\Phi_B)) \times \kappa(E(\Psi_A) \in U/IND(\Psi_A)) \times \kappa(E(\Phi_B) \in U/IND(\Phi_B))), \tag{34}$$

where

$$\kappa(E(\Psi_A) \subseteq E(\Phi_B)) = \begin{cases} 1 \text{ if } E(\Psi_A) \subseteq E(\Phi_B), \\ 0 \text{ otherwise.} \end{cases} \tag{35}$$

Proposition 6
If $(E(\Psi_A), \kappa(E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A)))$ in information table T is an element of $\underline{Apr}(\Phi_B, \Psi_A)$, there exists set $\underline{PT}$ of possible tables where for all $pt \in \underline{PT}$ $\overline{Apr}(\Phi_B, \Psi_A)_{pt}$ contains $E(\Psi_A)$ and $\sum_{pt\in PT} \mu(pt)$ is equal to $\kappa(E(\Psi_A) \in \underline{Apr}(\Phi_B, \Psi_A))$.

For the upper approximation,

$$\overline{Apr}(\Phi_B, \Psi_A) = \{(E(\Psi_A), \kappa(o \in \overline{Apr}(\Phi_B, \Psi_A))) \mid \kappa(E(\Psi_A) \in \overline{Apr}(\Phi_B, \Psi_A)) > 0\}, \tag{36}$$

$$\kappa(E(\Psi_A) \in \overline{Apr}(\Phi_B, \Psi_A)) = \kappa(E(\Psi_A) \cap \Phi_B \neq \emptyset) \times \kappa(E(\Psi_A) \in U/IND(\Psi_A)), \tag{37}$$

where

$$\kappa(E(\Psi_A) \cap \Phi_B \neq \emptyset) = \begin{cases} 1 \text{ if } E(\Psi_A) \cap \Phi_B \neq \emptyset, \\ 0 \text{ otherwise.} \end{cases} \tag{38}$$

From this formula, the upper approximation is trivial when $\Phi_B = U_B$; namely, $\overline{Apr}(U_B, \Psi_A) = U/IND(\Psi_A)$.

Proposition 7
If $(E(\Psi_A), \kappa(E(\Psi_A) \in \overline{Apr}(\Phi_B, \Psi_A)))$ in information table T is an element of $\overline{Apr}(\Phi_B, \Psi_A)$, there exists set $\overline{PT}$ of possible tables where for all $pt \in \overline{PT}$ $\overline{Apr}(\Phi_B, \Psi_A)_{pt}$ contains $E(\Psi_A)$ and $\sum_{pt \in PT} \mu(pt)$ is equal to $\kappa(E(\Psi_A) \in \overline{Apr}(\Phi_B, \Psi_A))$.

For expressions in terms of a set of objects, the same expressions as in Section 3 are used.

Proposition 8
The lower and upper approximations that are obtained by the method of weighted equivalence classes coincide with ones obtained by the method of possible worlds.

5 Information Tables Containing Missing Values

We apply the method of weighted equivalence classes to information tables containing missing values. We briefly compare the method where Kryszkiewicz uses indiscernible classes with the method of weighted equivalence classes.

When missing values are contained in information table T, Kryszkiewicz defines binary relation $IND(U_A)$ for indiscernibility between objects on set A of attributes as follows [8,10]:

$$IND(U_A) = \{(o, o') \in U \times U \mid \forall a \in A, a(o) = a(o') \vee a(o) = * \vee a(o') = *\}, \quad (39)$$

where $*$ denotes a missing value and U is used in place of Ψ when Ψ is equal to universe U. From this definition, an object having missing values for all attributes on set A of attributes is indiscernible with any object. This corresponds to "do not care" semantics of missing values addressed by Grzymala-Busse [4,5]. By using indiscernible classes obtained from $IND(U_A)$, Kryszkiewicz expresses lower and upper approximations of set $\Phi \subseteq U$ of objects:

$$\underline{apr}(\Phi, U_A) = \{o \in U \mid S_A(o) \subseteq \Phi\}, \quad (40)$$
$$\overline{apr}(\Phi, U_A) = \{o \in U \mid S_A(o) \cap \Phi \neq \emptyset\}, \quad (41)$$

where $S_A(o) (= \{o' \mid (o, o') \in IND(U_A)\})$ denotes the indiscernible class for object o.

When we use the method of weighted equivalence classes, a missing value in an attribute is probabilistically interpreted. In the missing value, every element in the domain of the attribute has the same probability to which the element is the actual value. In other words, a missing value in attribute a is equal to the probabilistic value expressed in the uniform probability distribution where every element over the domain has the same probability $1/|V_a|$. When attribute value $a(o)$ of object o is a missing value,

$$\kappa(a(o) = a(o')) = \sum_{u,v \in V_a} (\mu_{=}(u, v) \times \pi_{a(o)}(u) \times \pi_{a(o')}(v)) = 1/|V_a|,$$

where $\pi_{a(o)}(u)$ and $\pi_{a(o')}(u)$ denote probability distributions expressing attribute values $a(o)$ and $a(o')$,[3] respectively, and,

$$\mu_{=}(u,v) = \begin{cases} 1 \text{ if } u = v, \\ 0 \text{ otherwise.} \end{cases}$$

This shows that the indiscernibility degree of an object taking a missing value on attribute a with the other objects is equal to $1/|V_a|$; namely, the object is indiscernible with all objects with probability $1/|V_a|$. We express lower and upper approximations in terms of weighted equivalence classes, as is shown in the previous section. Differences between the method of Kryszkiewicz and the method of weighted equivalence classes are clarified in the following simple example:

Example
We suppose that information table T is obtained:

T

O	a_1	a_2	a_3
1	x	1	a
2	x	1	a
3	x	1	a
4	x	1	a
5	$*$	2	b

The mark O denotes the object identity. Domains V_{a_1}, V_{a_2}, and V_{a_3} of attributes a_1, a_2, and a_3 are $\{x, y\}$, $\{1, 2\}$, and $\{a, b\}$, respectively.

First, we apply the method of Kryszkiewicz to information table T. For indiscernible classes of each object on attribute a_1,

$$S_{a_1}(o_1) = S_{a_1}(o_2) = S_{a_1}(o_3) = S_{a_1}(o_4) = S_{a_1}(o_5) = \{o_1, o_2, o_3, o_4, o_5\}.$$

We suppose that $\Phi = \{o_1, o_2, o_3, o_4\}$ for simplicity. For the lower approximation, using formula (40), because of $\{o_1, o_2, o_3, o_4, o_5\} \not\subseteq \{o_1, o_2, o_3, o_4\}$,

$$\underline{apr}(\Phi, U_{a_1}) = \emptyset$$

This shows that we do not obtain any information for the lower approximation.[4] This is true for different expressions [4,6,12] proposed by several authors. For the upper approximation, using formula (41), because of $\{o_1, o_2, o_3, o_4, o_5\} \cap \{o_1, o_2, o_3, o_4\} \neq \emptyset$,

$$\overline{apr}(\Phi, U_{a_1}) = \{o_1, o_2, o_3, o_4, o_5\}.$$

Second, we use the method of weighted equivalence classes. Missing value $*$ in information table T is expressed in probability distribution $\{(x, 1/2), (y, 1/2)\}_p$. Using formulas (24) – (31),

[3] When $a(o')$ is a precise value; for example, $a(o') = x$, probability distribution $\pi_{a(o')}$ is expressed in $\{(x, 1)\}_p$, where subscript p denotes a probability distribution.

[4] Stefanowski and Tsoukiàs points out that the method of Kryszkiewicz using "do not care" semantics creates quite poor results [24]. To handle the problem, other assumptions for indiscernibility of missing values are proposed [2,24].

$$U/IND(U_{a_1}) = \{(\{o_1, o_2, o_3, o_4\}, 1/2), (\{o_1, o_2, o_3, o_4, o_5\}, 1/2)\},$$

Applying formulas (33) – (38),

$$\underline{\underline{Apr}}(\Phi, U_{a_1}) = \{(\{o_1, o_2, o_3, o_4\}, 1/2)\},$$
$$\overline{Apr}(\Phi, U_{a_1}) = \{(\{o_1, o_2, o_3, o_4\}, 1/2), \{(\{o_1, o_2, o_3, o_4, o_5\}, 1/2)\}.$$

Using formulas (18) – (21),

$$\underline{apr}(\Phi, U_{a_1}) = \{(o_1, 1/2), (o_2, 1/2), (o_3, 1/2), (o_4, 1/2)\},$$
$$\overline{apr}(\Phi, U_{a_1}) = \{(o_1, 1), (o_2, 1), (o_3, 1), (o_4, 1), (o_5, 1/2)\}.$$

Last, we show results by the method of possible worlds. Extended set $rep(T)$ of possible tables is,

$$rep(T) = \{(pt_1, 1/2), (pt_2, 1/2)\}_p.$$

pt_1

O	a_1	a_2	a_3
1	x	1	a
2	x	1	a
3	x	1	a
4	x	1	a
5	x	2	b

pt_2

O	a_1	a_2	a_3
1	x	1	a
2	x	1	a
3	x	1	a
4	x	1	a
5	y	2	b

For families of equivalence classes of possible tables,

$$(U/IND(U_{a_1}), 1/2)_{pt_1} = \{(\{o_1, o_2, o_3, o_4, o_5\}, 1/2)\},$$
$$(U/IND(U_{a_1}), 1/2)_{pt_2} = \{(\{o_1, o_2, o_3, o_4\}, 1/2), (\{o_5\}, 1/2)\},$$

For lower and upper approximations of each possible table,

$$\underline{\underline{Apr}}(\Phi, U_{a_1})_{pt_1} = \emptyset,$$
$$\overline{Apr}(\Phi, U_{a_1})_{pt_1} = \{(\{o_1, o_2, o_3, o_4, o_5\}, 1/2)\},$$
$$\underline{\underline{Apr}}(\Phi, U_{a_1})_{pt_2} = \{(\{o_1, o_2, o_3, o_4\}, 1/2)\}.$$
$$\overline{Apr}(\Phi, U_{a_1})_{pt_2} = \{(\{o_1, o_2, o_3, o_4\}, 1/2)\}.$$

Finally, using formulas (12) – (17) and (18) – (21),

$$\underline{\underline{Apr}}(\Phi, U_{a_1}) = \{(\{o_1, o_2, o_3, o_4\}, 1/2)\},$$
$$\overline{Apr}(\Phi, U_{a_1}) = \{(\{o_1, o_2, o_3, o_4\}, 1/2), (\{o_1, o_2, o_3, o_4, o_5\}, 1/2)\},$$
$$\underline{\underline{apr}}(\Phi, U_{a_1}) = \{(o_1, 1/2), (o_2, 1/2), (o_3, 1/2), (o_4, 1/2)\},$$
$$\overline{apr}(\Phi, U_{a_1}) = \{(o_1, 1), (o_2, 1), (o_3, 1), (o_4, 1), (o_5, 1/2)\}.$$

Indeed, the results obtained from the method of weighted equivalence classes coincide with ones from the method of possible worlds.

This simple example shows that we obtain correct results for the lower approximation when weighted equivalence classes are used. On the other hand, we cannot obtain any information for the lower approximation by the existence of only the missing value in the method of Kryszkiewicz where indiscernible classes are used.

6 Conclusions

We have proposed a method, where weighted equivalence classes are used, to deal with imprecise information expressed in a probability distribution. The lower and upper approximations by the method of weighted equivalence classes coincide with ones by the method of possible worlds. In other words, this method satisfies the correctness criterion that is used in the field of incomplete databases. This is justification of the method of weighted equivalence classes.

We have applied the method of weighted equivalence classes to information tables containing missing values under probabilistic interpretation. We obtain correct results for rough approximations when weighted equivalence classes are used, even if we do not obtain any results for the lower approximation when the method of Kryszkiewicz is used.

Acknowledgment. This research has been partially supported by the Grant-in-Aid for Scientific Research (C), Japan Society for the Promotion of Science, No. 18500214.

References

1. Abiteboul, S., Hull, R., Vianu, V.: Foundations of Databases. Addison-Wesley, London, UK (1995)
2. Greco, S., Matarazzo, B., Slowinski, R.: Handling Missing Values in Rough Set Analysis of Multi-attribute and Multi-criteria Decision Problem. In: Zhong, N., Skowron, A., Ohsuga, S. (eds.) New Directions in Rough Sets, Data Mining, and Granular-Soft Computing. LNCS (LNAI), vol. 1711, pp. 146–157. Springer, Heidelberg (1999)
3. Grzymala-Busse, J.W.: On the Unknown Attribute Values in Learning from Examples. In: Raś, Z.W., Zemankova, M. (eds.) ISMIS 1991. LNCS(LNAI), vol. 542, pp. 368–377. Springer, Heidelberg (1991)
4. Grzymala-Busse, J.W: Data with Missing Attribute Values: Generalization of Indiscernibility Relation and Rule Induction, Transactions on Rough Sets I, 78–95 (2004)
5. Grzymala-Busse, J.W.: Characteristic Relations for Incomplete Data: A Generalization of the Indiscernibility Relation, Transactions on Rough Sets IV, 58–68 (2005)
6. Guan, Y.-Y., Wang, H.-K.: Set-valued Information Systems. Information Sciences 176, 2507–2525 (2006)
7. Imielinski, T., Lipski, W.: Incomplete Information in Relational Databases. Journal of the ACM 31(4), 761–791 (1984)
8. Kryszkiewicz, M.: Rough Set Approach to Incomplete Information Systems. Information Sciences 112, 39–49 (1998)
9. Kryszkiewicz, M.: Properties of Incomplete Information Systems in the framework of Rough Sets. In: Polkowski, L., Skowron, A. (eds.) Rough Set in Knowledge Discovery 1: Methodology and Applications, Studies in Fuzziness and Soft Computing, vol. 18, pp. 422–450. Physica Verlag, Heidelberg (1998)
10. Kryszkiewicz, M.: Rules in Incomplete Information Systems. Information Sciences 113, 271–292 (1999)
11. Latkowski, R.: On Decomposition for Incomplete Data. Fundamenta Informaticae 54, 1–16 (2003)

12. Leung, Y., Li, D.: Maximum Consistent Techniques for Rule Acquisition in Incomplete Information Systems. Information Sciences 153, 85–106 (2003)
13. Nakata, N., Sakai, H.: Rough-set-based approaches to data containing incomplete information: possibility-based cases, pp. 234–241. IOS Press, Amsterdam, Trento, Italy (2005)
14. Nakata, N., Sakai, H.: Checking Whether or Not Rough-Set-Based Methods to Incomplete Data Satisfy a Correctness Criterion. In: Torra, V., Narukawa, Y., Miyamoto, S. (eds.) MDAI 2005. LNCS (LNAI), vol. 3558, pp. 227–239. Springer, Heidelberg (2005)
15. Nakata, N., Sakai, H.: Rough Sets Handling Missing Values Probabilistically Interpreted. In: Ślęzak, D., Wang, G., Szczuka, M., Düntsch, I., Yao, Y. (eds.) RSFDGrC 2005. LNCS (LNAI), vol. 3641, pp. 325–334. Springer, Heidelberg (2005)
16. Nakata, N., Sakai, H.: Lower and Upper Approximations in Data Tables Containing Possibilistic Information, Transactions on Rough Sets VII, 170–189 (2007)
17. Orłowska, E., Pawlak, Z.: Representation of Nondeterministic Information. Theoretical Computer Science 29, 313–324 (1984)
18. Parsons, S.: Current Approaches to Handling Imperfect Information in Data and Knowledge Bases. IEEE Transactions on Knowledge and Data Engineering 8(3), 353–372 (1996)
19. Pawlak, Z.: Rough Sets. International Journal of Computer and Information Sciences 11, 341–356 (1982)
20. Sakai, H.: Effective Procedures for Handling Possible Equivalence Relation in Nondeterministic Information Systems. Fundamenta Informaticae 48, 343–362 (2001)
21. Sakai, H., Nakata, M.: An Application of Discernibility Functions to Generating Minimal Rules in Non-deterministic Information Systems. Journal of Advanced Computational Intelligence and Intelligent Informatics 10, 695–702 (2006)
22. Sakai, H., Okuma, A.: Basic Algorithms and Tools for Rough Non-deterministic Information Systems, Transactions on Rough Sets I, 209–231 (2004)
23. Słowiński, R., Stefanowski, J.: Rough Classification in Incomplete Information Systems. Mathematical and Computer Modelling 12(10/11), 1347–1357 (1989)
24. Stefanowski, J., Tsoukiàs, A.: On the Extension of Rough Sets under Incomplete Information. In: Zhong, N., Skowron, A., Ohsuga, S. (eds.) New Directions in Rough Sets, Data Mining, and Granular-Soft Computing. LNCS (LNAI), vol. 1711, pp. 212–219. Springer, Heidelberg (1999)
25. Stefanowski, J., Tsoukiàs, A.: Incomplete Information Tables and Rough Classification. Computational Intelligence 17(3), 545–566 (2001)
26. Ziarko, W.: Probabilistic Rough Sets. In: Ślęzak, D., Wang, G., Szczuka, M., Düntsch, I., Yao, Y. (eds.) RSFDGrC 2005. LNCS (LNAI), vol. 3641, pp. 283–293. Springer, Heidelberg (2005)
27. Ziarko, W.: Stochastic Approach to Rough Set Theory. In: Greco, S., Hata, Y., Hirano, S., Inuiguchi, M., Miyamoto, S., Nguyen, H.S., Słowiński, R. (eds.) RSCTC 2006. LNCS (LNAI), vol. 4259, pp. 38–48. Springer, Heidelberg (2006)
28. Zimányi, E., Pirotte, A.: Imperfect Information in Relational Databases. In: Motro, A., Smets, P. (eds.) Uncertainty Management in Information Systems: From Needs to Solutions, pp. 35–87. Kluwer Academic Publishers, Boston, MA (1997)

Fuzzy Extension of Estimations with Randomness: The Perception-Based Approach

Yuji Yoshida

Faculty of Economics and Business Administration, University of Kitakyushu
4-2-1 Kitagata, Kokuraminami, Kitakyushu 802-8577, Japan
yoshida@kitakyu-u.ac.jp

Abstract. A set of perceived random events is given by a fuzzy random variable, and its estimation is represented by a functional on real random variables. The estimation of the perception regarding random events is obtained, extending the functional to a functional of fuzzy random variables. This paper discusses conditions and various properties of perception-based extension of estimations with randomness, and several examples of the perception-based extension are investigated. The results can be applied other estimations in engineering, economics and so on.

1 Introduction

Fuzzy random variables, which were introduced by Kwakernaak [7,8], are applied to decision-making under uncertainty with fuzziness like linguistic data in statistics, engineering and economics. The notion of fuzzy random variable is strongly related to perception of random events. The relation between fuzzy logic with randomness and perception is found in Zadeh [17]. On the other hand, in decision-making with randomness we need estimations of real random variables like the expectation as a criterion of optimization. This paper discusses conditions and properties of an extended estimation for fuzzy random variables. In this paper, a set of perceived random events, which are represented by real random variables, is given by a fuzzy random variable. An estimation of perceived random events is also represented by a functional on real random variables. Then, we obtain the estimation of the perception regarding random events, extending the functional to a functional of fuzzy random variables. We deal with a fuzzy random variable and its estimation. In the rest of this section, we explain the details with mathematical notations.

Put the set of all real numbers by $\mathbb{R} := (-\infty, \infty)$. A *fuzzy number* is denoted by its membership function $\tilde{a} : \mathbb{R} \mapsto [0, 1]$ which is normal, upper-semicontinuous, fuzzy convex and has a compact support (Zadeh [16]). $\mathcal{R}$ denotes the set of all fuzzy numbers. Let Ω be a sample space and let $\mathcal{X}$ be a set of real random variables on Ω. A fuzzy-number-valued map $\tilde{X} : \Omega \mapsto \mathcal{R}$ is called a *fuzzy random variable* if $\omega \mapsto \tilde{X}_\alpha^-(\omega)$ and $\omega \mapsto \tilde{X}_\alpha^+(\omega)$ are measurable for all $\alpha \in [0, 1]$, where $\tilde{X}_\alpha(\omega) = [\tilde{X}_\alpha^-(\omega), \tilde{X}_\alpha^+(\omega)] := \{x \in \mathbb{R} \mid \tilde{X}(\omega)(x) \geq \alpha\}$ is the α-cut (Kwakernaak [7,8]). Kruse and Meyer [5] gave a *perception-based definition regarding*

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 295–306, 2007.

the expectation of a fuzzy random variable $\tilde{X}$, which is induced from Zadeh's extension principle, as follows.

$$\theta(\tilde{X})(x) := \sup_{X\in\mathcal{X}\,:\,E(X)=x} \inf_{\omega\in\Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R}, \tag{1}$$

where $\mathcal{X}$ is taken as the set of all integrable real random variables and $E(X) := \int X dP$. Then, it is known that the expectation $\theta(\tilde{X})$ is a fuzzy number whose α-cut is given by

$$\theta(\tilde{X})_\alpha = [E(\tilde{X}_\alpha^-), E(\tilde{X}_\alpha^+)] \tag{2}$$

for $\alpha \in (0, 1]$. Hence, the term $\inf_{\omega\in\Omega} \tilde{X}(\omega)(X(\omega))$ in (1) means a level at which we can perceive the real random variable $X(\omega)$ by the fuzzy random variable $\tilde{X}(\omega)(\cdot)$ since actually for each level $\alpha \in (0, 1]$ it holds that $\inf_{\omega\in\Omega} \tilde{X}(\omega)(X(\omega)) \geq \alpha$ if and only if $X(\omega) \in \tilde{X}_\alpha(\omega)$ for all $\omega \in \Omega$. Therefore, the definition (1) implies that the expectation of a fuzzy random variable $\tilde{X}$ is obtained by applying Zadeh's extension principle to the expectation operation $E(\cdot)$ of real random variables $X(\omega)$ which we can perceive by the fuzzy random variable $\tilde{X}(\omega)(\cdot)$. The α-cut of the fuzzy number (1) can be generally given by the following *Aumann integral*:

$$\theta(\tilde{X})_\alpha = \{E(X) | X \in \mathcal{X} \text{ and } X(\omega) \in \tilde{X}_\alpha(\omega) \text{ for all } \omega \in \Omega\}. \tag{3}$$

Puri and Ralescu [11] discussed the conditional expectation of fuzzy random variables by Aumann integral, and López-Díaz et al. [2] discussed it for the study of statistics with fuzzy data. In this paper, for a functional $\varphi : \mathcal{X} \mapsto \mathcal{R}$ as a more general estimation of real random variables, by the perception-based approach we discuss *fuzzy extensions* $\tilde{\varphi}$ of the estimation φ which is defined in the following:

$$\tilde{\varphi}(\tilde{X})(x) := \sup_{X\in\mathcal{X}\,:\,\varphi(X)=x} \inf_{\omega\in\Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R} \tag{4}$$

for a fuzzy random variable $\tilde{X} \in \tilde{\mathcal{X}}$. Refer to Fig.1 for the image of the extension.

In the next section, we discuss some conditions and various properties regarding perception-based extension (4) of estimations with randomness. In Section 3,

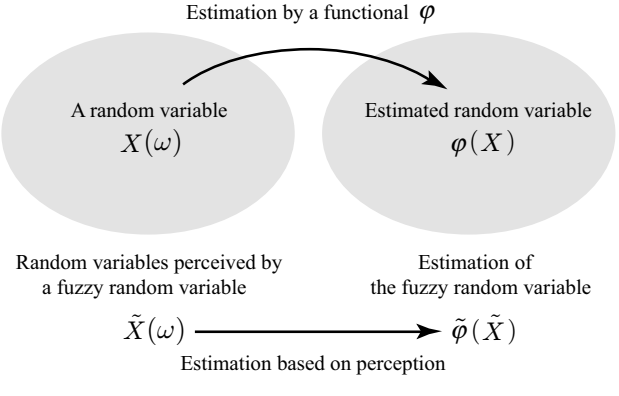


Fig. 1. Estimation of a fuzzy random variable based on perception

we investigate several examples of the perception-based extension in statistics, engineering and finance. In the last section, we discuss an approach to cases where the monotone condition, which is given in Section 2, does not hold.

2 Perception-Based Extension of Estimations with Randomness

For a fuzzy number $\tilde{a}(\in \mathcal{R})$, its α-cuts are given by $\tilde{a}_\alpha := \{x \in \mathbb{R} \mid \tilde{a}(x) \geq \alpha\}$ $(\alpha \in (0,1])$ and $\tilde{a}_0 := \mathrm{cl}\{x \in \mathbb{R} \mid \tilde{a}(x) > 0\}$, where cl denotes the closure of an interval. The α-cut is also written by closed intervals $\tilde{a}_\alpha = [\tilde{a}_\alpha^-, \tilde{a}_\alpha^+]$ $(\alpha \in [0,1])$. Hence we introduce a partial order $\succeq$, so called the *fuzzy max order*, on fuzzy numbers $\mathcal{R}$: Let $\tilde{a}, \tilde{b} \in \mathcal{R}$ be fuzzy numbers. $\tilde{a} \succeq \tilde{b}$ means that $\tilde{a}_\alpha^- \geq \tilde{b}_\alpha^-$ and $\tilde{a}_\alpha^+ \geq \tilde{b}_\alpha^+$ for all $\alpha \in [0,1]$. An addition and a scalar multiplication for fuzzy numbers are defined as follows: For $\tilde{a}, \tilde{b} \in \mathcal{R}$ and $\lambda \in \mathbb{R}$, the addition $\tilde{a} + \tilde{b}$ of $\tilde{a}$ and $\tilde{b}$ and the scalar multiplication $\lambda\tilde{a}$ of λ and $\tilde{a}$ are fuzzy numbers given by their α-cuts $(\tilde{a} + \tilde{b})_\alpha := [\tilde{a}_\alpha^- + \tilde{b}_\alpha^-, \tilde{a}_\alpha^+ + \tilde{b}_\alpha^+]$ and $(\lambda\tilde{a})_\alpha := [\lambda\tilde{a}_\alpha^-, \lambda\tilde{a}_\alpha^+]$ if $\lambda \geq 0$ and $(\lambda\tilde{a})_\alpha := [\lambda\tilde{a}_\alpha^+, \lambda\tilde{a}_\alpha^-]$ if $\lambda < 0$, where $\tilde{a}_\alpha = [\tilde{a}_\alpha^-, \tilde{a}_\alpha^+]$ and $\tilde{b}_\alpha = [\tilde{b}_\alpha^-, \tilde{b}_\alpha^+]$ $(\alpha \in [0,1])$. Now we also use the following *metric* d_H on $\mathcal{R}$ induced from Hausdorff metric: $d_H(\tilde{a}, \tilde{b}) := \sup_{\alpha\in[0,1]} \max\{|\tilde{a}_\alpha^- - \tilde{b}_\alpha^-|, |\tilde{a}_\alpha^+ - \tilde{b}_\alpha^+|\}$ for $\tilde{a}, \tilde{b} \in \mathcal{R}$.

Let $\mathcal{X}$ be a set of real random variables on a sample space Ω such that $\lambda X + \nu Y \in \mathcal{X}$ for all $X, Y \in \mathcal{X}$ and $\lambda, \nu \in \mathbb{R}$. Let $\tilde{\mathcal{X}}$ be a set of fuzzy random variables $\tilde{X}$ on Ω such that $\tilde{X}_\alpha^\pm \in \mathcal{X}$ for all $\alpha \in [0,1]$. We consider the following conditions regarding functionals $\varphi : \mathcal{X} \mapsto \mathbb{R}$.

- $(\varphi, \leq)$ $\varphi(X) \leq \varphi(Y)$ holds for all real random variables $X, Y \in \mathcal{X}$ satisfying $X(\omega) \leq Y(\omega)$ for all $\omega \in \Omega$. (non-decreasing property)
- $(\varphi, \geq)$ $\varphi(X) \geq \varphi(Y)$ holds for all real random variables $X, Y \in \mathcal{X}$ satisfying $X(\omega) \leq Y(\omega)$ for all $\omega \in \Omega$. (non-increasing property)
- (φ, c) Let $\{X_n\}_n \subset \mathcal{X}$ and $X \in \mathcal{X}$ be a sequence and its limit, i.e. $\lim_{n\to\infty} X_n(\omega) = X(\omega)$ for all $\omega \in \Omega$. Then it holds that $\lim_{n\to\infty} \varphi(X_n) = \varphi(X)$. (continuity)

The properties $(\varphi, \leq)$ and $(\varphi, \geq)$ are called *monotonicity*. We put a family of functionals with the non-decreasing property $(\varphi, \leq)$ by $\Phi^\leq := \{\text{functionals}\, \varphi : \mathcal{X} \mapsto \mathbb{R}\, \text{satisfying}\, (\varphi, \leq) \text{ and } (\varphi, \mathrm{c})\}$, and we also put a family of functionals with the non-increasing property $(\varphi, \geq)$ by $\Phi^\geq := \{\text{functionals}\, \varphi : \mathcal{X} \mapsto \mathbb{R}\, \text{satisfying}\, (\varphi, \leq) \text{ and } (\varphi, \mathrm{c})\}$.

Proposition 1. *For the families $\Phi^\leq$ and $\Phi^\geq$, the following (i) – (iii) hold.*

(i) *If $\varphi^1, \varphi^2 \in \Phi^\leq$, then $\varphi^1 + \varphi^2 \in \Phi^\leq$. If $\varphi^1, \varphi^2 \in \Phi^\geq$, then $\varphi^1 + \varphi^2 \in \Phi^\geq$.*
(ii) *Let $\varphi \in \Phi^\leq$. Then, $\lambda\varphi \in \Phi^\leq$ for $\lambda \geq 0$, and $\lambda\varphi \in \Phi^\geq$ for $\lambda \leq 0$.*
(iii) *Let $\varphi \in \Phi^\geq$. Then, $\lambda\varphi \in \Phi^\geq$ for $\lambda \geq 0$, and $\lambda\varphi \in \Phi^\leq$ for $\lambda \leq 0$.*

Proof. The proof of this proposition is trivial from the definitions. □

Now we obtain the following results regarding the families $\Phi^{\leq}$ and $\Phi^{\geq}$. Lemma 1 is derived from the properties $(\varphi, \leq)$, $(\varphi, \geq)$ and (φ, c), and it is used in the proof of Theorem 1.

Lemma 1. *Let $X, Y \in \mathcal{X}$ be real random variables satisfying $X \leq Y$. Put $X^\lambda = \lambda X + (1-\lambda)Y$ for $\lambda \in [0,1]$. If $\varphi \in \Phi^{\leq}$ ($\varphi \in \Phi^{\geq}$), then for any $x \in [\varphi(X), \varphi(Y)]$ ($x \in [\varphi(Y), \varphi(X)]$ resp.) there exists $\lambda (\in [0,1])$ such that $\varphi(X^\lambda) = x$.*

Proof. It is sufficient to prove the case of $\varphi \in \Phi^{\leq}$. Let $x \in [\varphi(X), \varphi(Y)]$. In case of $x = \varphi(X)$ ($x = \varphi(Y)$), we take $\lambda = 1$ ($\lambda = 0$ resp.). Therefore we consider only cases where $\varphi(X) < x < \varphi(Y)$. We have $X^\lambda = \lambda X + (1-\lambda)Y \in \mathcal{X}$ for $\lambda \in [0,1]$. Suppose that there exist no $\lambda (\in [0,1])$ such that $\varphi(X^\lambda) = x$. Put $\Lambda^- := \{\lambda \in [0,1] | \varphi(X^\lambda) < x\}$ and $\Lambda^+ := \{\lambda \in [0,1] | \varphi(X^\lambda) > x\}$. Then $\Lambda^- \cup \Lambda^+ = [0,1]$. From the property $(\varphi, \leq)$, there exists $\lambda^* (\in [0,1])$ such that $\varphi(X^\lambda) < x$ for all $\lambda \in (\lambda^*, 1]$ and $\varphi(X^\lambda) > x$ for all $\lambda \in [0, \lambda^*)$. Then we have $X^{\lambda^*} = \lim_{\lambda \to \lambda^*} X^\lambda$. From the property (φ, c), it follows $\varphi(X^{\lambda^*}) = \lim_{\lambda \to \lambda^*} \varphi(X^\lambda)$. By the definition of λ^*, we also have $\lim_{\lambda \uparrow \lambda^*} \varphi(X^\lambda) \geq x$ and $\lim_{\lambda \downarrow \lambda^*} \varphi(X^\lambda) \leq x$. These imply $\varphi(X^{\lambda^*}) = \lim_{\lambda \to \lambda^*} \varphi(X^\lambda) = x$, and this equation contradicts the definition of λ^*. Thus we obtain this lemma. □

Theorem 1. *Let $\varphi \in \Phi^{\leq} \cup \Phi^{\geq}$. Define a functional $\tilde{\varphi} : \tilde{\mathcal{X}} \mapsto \mathcal{R}$ by*

$$\tilde{\varphi}(\tilde{X})(x) := \sup_{X \in \mathcal{X} : \varphi(X) = x} \inf_{\omega \in \Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R} \tag{5}$$

for a fuzzy random variable $\tilde{X} \in \tilde{\mathcal{X}}$. If $\varphi \in \Phi^{\leq}$ ($\varphi \in \Phi^{\geq}$), then the α-cut of $\tilde{\varphi}(\tilde{X})$ is given by a closed interval

$$\tilde{\varphi}(\tilde{X})_\alpha = [\varphi(\tilde{X}^-_\alpha), \varphi(\tilde{X}^+_\alpha)] \quad (\tilde{\varphi}(\tilde{X})_\alpha = [\varphi(\tilde{X}^+_\alpha), \varphi(\tilde{X}^-_\alpha)] \text{ resp.}) \tag{6}$$

for $\alpha \in (0,1]$.

Proof. It is sufficient to prove the case of $\varphi \in \Phi^{\leq}$. Let $\tilde{X} \in \tilde{\mathcal{X}}$ be a fuzzy random variable and let $\alpha \in (0,1]$. Now we check $\varphi(\tilde{X})_\alpha \subset [\varphi(\tilde{X}^-_\alpha), \varphi(\tilde{X}^+_\alpha)]$. Take a point x such that $x \in \varphi(\tilde{X})_\alpha$. Hence for any $\varepsilon > 0$ there exists $X \in \mathcal{X}$ such that $\varphi(X) = x$ and

$$\tilde{X}(\omega)(X(\omega)) \geq \alpha - \varepsilon \quad \text{for all } \omega \in \Omega.$$

This follows $X(\omega) \in \tilde{X}(\omega)_{\alpha-\varepsilon}$ for all $\omega \in \Omega$, and so we obtain

$$\tilde{X}(\omega)^-_{\alpha-\varepsilon} \leq X(\omega) \leq \tilde{X}(\omega)^+_{\alpha-\varepsilon} \quad \text{for all } \omega \in \Omega.$$

From the property $(\varphi, \leq)$, for any $\varepsilon > 0$ there exists $X \in \mathcal{X}$ such that $\varphi(X) = x$ and

$$\varphi(\tilde{X}^-_{\alpha-\varepsilon}) \leq \varphi(X) \leq \varphi(\tilde{X}^+_{\alpha-\varepsilon}).$$

Letting $\varepsilon \to 0$, by the property (φ, c) it follows

$$\varphi(\tilde{X}^-_\alpha) \leq x \leq \varphi(\tilde{X}^+_\alpha).$$

Thus we get $x \in [\varphi(\tilde{X}_\alpha^-), \varphi(\tilde{X}_\alpha^+)]$, and it holds that $\varphi(\tilde{X})_\alpha \subset [\varphi(\tilde{X}_\alpha^-), \varphi(\tilde{X}_\alpha^+)]$. On the contrary, we suppose $x \in [\varphi(\tilde{X}_\alpha^-), \varphi(\tilde{X}_\alpha^+)]$ and $\varphi(\tilde{X}_\alpha^-) < \varphi(\tilde{X}_\alpha^+)$. Then

$$\varphi(\tilde{X}_\alpha^-) \le x \le \varphi(\tilde{X}_\alpha^+).$$

Put $X^\lambda = \lambda \tilde{X}_\alpha^- + (1-\lambda)\tilde{X}_\alpha^+ (\in \mathcal{X})$ for $\lambda \in [0,1]$. From the properties $(\varphi, \le)$ and (φ, c), the real valued function

$$\lambda (\in [0,1]) \mapsto \varphi(X^\lambda) (\in [\varphi(\tilde{X}_\alpha^-), \varphi(\tilde{X}_\alpha^+)])$$

is non-decreasing, continuous and onto. Therefore, by Lemma 1 there exists $\lambda (\in [0,1])$ such that $\varphi(X^\lambda) = x$. Thus we have $\varphi(X^\lambda) = x$ and

$$\tilde{X}_\alpha^-(\omega) \le X^\lambda(\omega) \le \tilde{X}_\alpha^+(\omega) \quad \text{for all } \omega \in \Omega.$$

This follows $X^\lambda(\omega) \in \tilde{X}_\alpha(\omega)$ for all $\omega \in \Omega$. and so

$$\tilde{X}(\omega)(X^\lambda(\omega)) \ge \alpha \quad \text{for all } \omega \in \Omega.$$

Namely

$$\tilde{\varphi}(\tilde{X})(x) = \sup_{X \in \mathcal{X} : \varphi(X) = x} \inf_{\omega \in \Omega} \tilde{X}(\omega)(X(\omega)) \ge \alpha.$$

Thus we obtain $x \in \varphi(\tilde{X})_\alpha$. Finally we also check the case $x \in [\varphi(\tilde{X}_\alpha^-), \varphi(\tilde{X}_\alpha^+)]$ and $\varphi(\tilde{X}_\alpha^-) = \varphi(\tilde{X}_\alpha^+)$. In this case, taking $X = \tilde{X}_\alpha^- (\in \mathcal{X})$, we can derive the same conclusion in the same argument. Therefore, we obtain $\varphi(\tilde{X})_\alpha = [\varphi(\tilde{X}_\alpha^-), \varphi(\tilde{X}_\alpha^+)]$ in the case of $\varphi \in \Phi^{\le}$. We obtain the results in case of $\Phi^{\ge}$ in the same way. □

The following proposition shows that the extended estimations $\tilde{\varphi}$ preserve the properties $(\varphi, \le)$, $(\varphi, \ge)$ and (φ, c).

Proposition 2. *The following (i) and (ii) hold.*

(i) *Let $\varphi \in \Phi^{\le}$ ($\varphi \in \Phi^{\ge}$). Then $\tilde{\varphi}(\tilde{X}) \preceq \tilde{\varphi}(\tilde{Y})$ ($\tilde{\varphi}(\tilde{X}) \succeq \tilde{\varphi}(\tilde{Y})$ resp.) holds for all fuzzy random variables $X, Y \in \tilde{\mathcal{X}}$ satisfying $\tilde{X}(\omega) \preceq \tilde{Y}(\omega)$ for all $\omega \in \Omega$, where $\preceq$ is the fuzzy max order on $\mathcal{R}$.*

(ii) *Let $\varphi \in \Phi^{\le} \cup \Phi^{\ge}$. Let $\{\tilde{X}_n\}_n \subset \tilde{\mathcal{X}}$ and $\tilde{X} \in \tilde{\mathcal{X}}$ be a sequence and its limit, i.e. $\lim_{n\to\infty} \tilde{X}_n(\omega) = \tilde{X}(\omega)$ for all $\omega \in \Omega$. Then it holds that $\lim_{n\to\infty} \tilde{\varphi}(\tilde{X}_n) = \tilde{\varphi}(\tilde{X})$, where we use the metric d_H on $\mathcal{R}$.*

Proof. We can easily check the proof of this proposition by Theorem 1 and the bounded convergence theorem. □

Finally we discuss the additivity and scalar multiplicativity for functionals $\varphi \in \Phi^{\le} \cup \Phi^{\ge}$. Real random variables $X, Y \in \mathcal{X}$ are called *comonotonic* if there exist strictly increasing functions $f, g : \mathbb{R} \mapsto \mathbb{R}$ and a real random variable $Z \in \mathcal{X}$ such that $X = f(Z)$ and $Y = g(Z)$. Fuzzy random variables $\tilde{X}, \tilde{Y} \in \tilde{\mathcal{X}}$ are also

called *comonotonic* if there exist strictly increasing functions $f, g : \mathbb{R} \mapsto \mathbb{R}$ and a fuzzy random variable $\tilde{Z} \in \tilde{\mathcal{X}}$ such that $\tilde{X} = f(\tilde{Z})$ and $\tilde{Y} = g(\tilde{Z})$, where

$$f(\tilde{Z})(\omega)(x) := \sup_{Z \in \mathcal{X} \,:\, f(Z(\omega)) = x} \tilde{Z}(\omega)(Z(\omega)), \tag{7}$$

$$g(\tilde{Z})(\omega)(x) := \sup_{Z \in \mathcal{X} \,:\, g(Z(\omega)) = x} \tilde{Z}(\omega)(Z(\omega)) \tag{8}$$

for $\omega \in \Omega$ and $x \in \mathbb{R}$.

Proposition 3. *For $\lambda \in \mathbb{R}$ and $\varphi \in \Phi^{\leq} \cup \Phi^{\geq}$, the following (i) – (iii) hold.*

(i) *If φ satisfies $\varphi(X+Y) \leq \varphi(X) + \varphi(Y)$ ($\varphi(X+Y) \geq \varphi(X) + \varphi(Y)$) for all $X, Y \in \mathcal{X}$, then it holds that $\tilde{\varphi}(\tilde{X} + \tilde{Y}) \preceq \tilde{\varphi}(\tilde{X}) + \tilde{\varphi}(\tilde{Y})$ ($\tilde{\varphi}(\tilde{X} + \tilde{Y}) \succeq \tilde{\varphi}(\tilde{X}) + \tilde{\varphi}(\tilde{Y})$ resp.) for all $\tilde{X}, \tilde{Y} \in \tilde{\mathcal{X}}$, where*

$$(\tilde{X} + \tilde{Y})(\omega)(x) := \sup_{X, Y \in \mathcal{X} \,:\, X(\omega) + Y(\omega) = x} \min\{\tilde{X}(\omega)(X(\omega)), \tilde{Y}(\omega)(Y(\omega))\} \tag{9}$$

for $\omega \in \Omega$ and $x \in \mathbb{R}$.

(ii) *If φ satisfies $\varphi(\lambda X) = \lambda \varphi(X)$ for all $X \in \mathcal{X}$, then it holds that $\tilde{\varphi}(\lambda \tilde{X}) = \lambda \tilde{\varphi}(\tilde{X})$ for all $\tilde{X} \in \tilde{\mathcal{X}}$, where*

$$(\lambda \tilde{X})(\omega)(x) := \sup_{X \in \mathcal{X} \,:\, \lambda X(\omega) = x} \tilde{X}(\omega)(X(\omega)) \tag{10}$$

for $\omega \in \Omega$ and $x \in \mathbb{R}$.

(iii) *If φ satisfies $\varphi(X+Y) = \varphi(X) + \varphi(Y)$ for all comonotonic real random variables $X, Y \in \mathcal{X}$, then it holds that $\tilde{\varphi}(\tilde{X} + \tilde{Y}) = \tilde{\varphi}(\tilde{X}) + \tilde{\varphi}(\tilde{Y})$ for all comonotonic fuzzy random variables $\tilde{X}, \tilde{Y} \in \tilde{\mathcal{X}}$.*

Proof. We can easily check the proof from Theorem 1. □

The property (iii) in Proposition 3 is called *comonotonic additive*.

3 Applications of Perception-Based Extension of Estimations

In this section, we give several examples of estimations, which are used in statistics, engineering and economics, and we investigate the main results in the previous section about the examples.

3.1 Expectations and Conditional Expectations

Let P be a non-atomic probability on a sample Ω. Let $\mathcal{X}$ be the set of all integrable real random variables on (Ω, P). A fuzzy random variable $\tilde{X}$ is called integrable if the both end of α-cuts $\tilde{X}_\alpha^{\pm}$ are integrable for all $\alpha \in [0, 1]$. Let $\tilde{\mathcal{X}}$

be the set of all integrable fuzzy random variables on (Ω, P). The *expectation* of an integrable fuzzy random variable $\tilde{X}$ is a fuzzy number

$$\tilde{E}(\tilde{X})(x) := \sup_{X\in\mathcal{X}\,:\,E(X)=x} \inf_{\omega\in\Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R}, \tag{11}$$

where we take $\varphi = E(\cdot) \in \Phi^{\leq}$ in (4) (Kruse and Meyer [5]). This extension is well-defined and has monotone, continuous and linear properties in Propositions 2 and 3.

Let $\mathcal{M}$ be a σ-field on Ω and let $\mathcal{G}$ be a sub-σ-field of $\mathcal{M}$. The *conditional expectation* of an integrable fuzzy random variable $\tilde{X}(\in \tilde{\mathcal{X}})$ is a fuzzy random variable

$$\tilde{E}(\tilde{X}|\mathcal{G})(\omega')(x) := \sup_{X\in\mathcal{X}\,:\,E(X|\mathcal{G})(\omega')=x} \inf_{\omega\in\Omega} \tilde{X}(\omega)(X(\omega)), \quad \omega' \in \Omega,\ x \in \mathbb{R}, \tag{12}$$

where we take $\varphi = E(\cdot|\mathcal{G})(\omega') \in \Phi^{\leq}$ in (4). Its α-cut is given by $\tilde{E}(\tilde{X}|\mathcal{G})_\alpha = [E(\tilde{X}_\alpha^-|\mathcal{G}), E(\tilde{X}_\alpha^+|\mathcal{G})]$, and this extension has non-decreasing, continuous and linear properties in Propositions 2 and 3. Puri and Ralescu [11] discussed the properties of this extension.

3.2 Cumulative Distribution Functions and Threshold Probabilities

Let P be a non-atomic probability on a sample Ω. Let $\mathcal{X}$ be the set of all real random variables on (Ω, P). Let $\tilde{\mathcal{X}}$ be the set of all fuzzy random variables on (Ω, P). The *cumulative distribution function* of a fuzzy random variable $\tilde{X}(\in \tilde{\mathcal{X}})$ is a fuzzy number

$$\tilde{P}(\tilde{X} < t)(x) := \sup_{X\in\mathcal{X}\,:\,P(X<t)=x} \inf_{\omega\in\Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R}, \tag{13}$$

where we take $\varphi = P(\ \cdot\ < t) \in \Phi^{\geq}$ in (4). Its α-cut is given by $\tilde{P}(\tilde{X} < t)_\alpha = [P(\tilde{X}_\alpha^+ < t), P(\tilde{X}_\alpha^- < t)]$, and this extension has non-increasing and continuous properties in Proposition 2. Cumulative distribution functions $P(X < t)$ are used in decision-making for the optimization of *threshold probabilities* (Ohtsubo [10]).

3.3 Quantiles, Median and Value-at-Risk

Let P be a non-atomic probability on a sample Ω. Let $\mathcal{X}$ be the set of real random variables X on (Ω, P) with a continuous cumulative distribution function $x \mapsto F_X(x) := P(X < x)$ for which there exists a non-empty open interval I such that $F_X(\cdot) : I \mapsto (0, 1)$ is a strictly increasing and onto. We put $F_X(\inf I) := 0$ and $F_X(\sup I) := 1$. Let the inverse function of F_X by $F_X^{-1}(\cdot) : [0, 1] \mapsto \mathrm{cl}\, I$, where $\mathrm{cl}\, I := I \cup \{\inf I\} \cup \{\sup I\}$ is the closure of I. Then *p-quantiles* are given by $F_X^{-1}(p)$ for probabilities $p \in [0, 1]$. Especially, *quantiles* $F_X^{-1}(0.25), F_X^{-1}(0.75)$ and the *median* $F_X^{-1}(0.5)$ are widely used in statistics (Meucci [9]).

In finance, the *Value-at-Risk (VaR)* at a probability level $p(\in [0, 1])$ is given by the p-quantile of the distribution function F_X as follows: $\mathrm{VaR}_p(X) := F_X^{-1}(p)$.

Let $\tilde{\mathcal{X}}$ be the set of all fuzzy random variables on (Ω, P). Then, Value-at-Risk of of a fuzzy random variable $\tilde{X}(\in \tilde{\mathcal{X}})$ is a fuzzy number

$$\widetilde{\mathrm{VaR}_p}(\tilde{X})(x) := \sup_{X\in\mathcal{X}\,:\,\mathrm{VaR}_p(X)=x} \inf_{\omega\in\Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R}, \tag{14}$$

where we take $\varphi = \mathrm{VaR}_p(\cdot) \in \Phi^{\leq}$ in (4). Its α-cut is given by $\widetilde{\mathrm{VaR}_p}(\tilde{X})_\alpha = [\mathrm{VaR}_p(\tilde{X}_\alpha^-), \mathrm{VaR}_p(\tilde{X}_\alpha^+)]$, and this extension has non-decreasing, positively homogeneous, continuous and comonotonic additive properties in Propositions 2 and 3.

3.4 Average Value-at-Risk / Expected Shortfall

Let P be a non-atomic probability on a set Ω. Let $\mathcal{X}$ be the set of real random variables X on (Ω, P) with a continuous cumulative distribution function on Ω. In finance, the *Average Value-at-Risk (AVaR) / Expected Shortfall (ES)* at a probability level $p(\in [0,1])$ is given as follows (Rockafellar and Uryasev [12], Tasche [13]).

$$\mathrm{AVaR}_p(X) := \begin{cases} \inf I & \text{if } p = 0 \\ \dfrac{1}{p}\displaystyle\int_0^p \mathrm{VaR}_q(X)\,dq = \frac{1}{p}\int_0^p F_X^{-1}(q)\,dq & \text{if } 0 < p \leq 1. \end{cases} \tag{15}$$

Let $\tilde{\mathcal{X}}$ be the set of all fuzzy random variables on (Ω, P). Then, Average Value-at-Risk / Expected Shortfall of a fuzzy random variable $\tilde{X}(\in \tilde{\mathcal{X}})$ is a fuzzy number

$$\widetilde{\mathrm{AVaR}_p}(\tilde{X})(x) := \sup_{X\in\mathcal{X}\,:\,\mathrm{AVaR}_p(X)=x} \inf_{\omega\in\Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R}, \tag{16}$$

where we take $\varphi = \mathrm{AVaR}_p(\cdot) \in \Phi^{\leq}$ in (4). Its α-cut is given by $\widetilde{\mathrm{AVaR}_p}(\tilde{X})_\alpha = [\mathrm{AVaR}_p(\tilde{X}_\alpha^-), \mathrm{AVaR}_p(\tilde{X}_\alpha^+)] = [\frac{1}{p}\int_0^p \mathrm{VaR}_q(\tilde{X}_\alpha^-)\,dq, \frac{1}{p}\int_0^p \mathrm{VaR}_q(\tilde{X}_\alpha^+)\,dq]$ for $p > 0$, and this extension is also well-defined and has non-decreasing, positively homogeneous, continuous and comonotonic additive properties in Propositions 2 and 3.

3.5 Call Options, Put Options and Expected Rewards

Let P be a non-atomic probability on a set Ω. Let $\mathcal{X}$ be the set of all integrable real random variables on (Ω, P). Let $\tilde{\mathcal{X}}$ be the set of all integrable fuzzy random variables on (Ω, P). In financial engineering, *call options* and *put options* of a stock price, which are defined by an integrable fuzzy random variable $\tilde{X}(\in \tilde{\mathcal{X}})$ with a strike price $\kappa(> 0)$ are given respectively by

$$\tilde{E}((\kappa - \tilde{X})^+)(x) := \sup_{X\in\mathcal{X}\,:\,E((\kappa-X)^+)=x} \inf_{\omega\in\Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R}, \tag{17}$$

$$\tilde{E}((\tilde{X} - \kappa)^+)(x) := \sup_{X\in\mathcal{X}\,:\,E((X-\kappa)^+)=x} \inf_{\omega\in\Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R}. \tag{18}$$

More generally, for continuous non-increasing/non-decreasing functions $f : \mathbb{R} \mapsto \mathbb{R}$, the *expected reward* of an integrable fuzzy random variable $\tilde{X}(\in \tilde{\mathcal{X}})$ is a fuzzy number

$$\tilde{E}(f(\tilde{X}))(x) := \sup_{X \in \mathcal{X} : E(f(X))=x} \inf_{\omega \in \Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R}, \tag{19}$$

where

$$f(\tilde{X})(\omega)(x) := \sup_{X \in \mathcal{X} : f(X(\omega))=x} \tilde{X}(\omega)(X(\omega)) \tag{20}$$

for $\omega \in \Omega$ and $x \in \mathbb{R}$. Hence we take $\varphi = E(f(\cdot)) \in \Phi^{\leq}$ in (4) for non-decreasing f and its α-cut is given by $\tilde{E}(f(\tilde{X}))_\alpha = [E(f(\tilde{X}_\alpha^-)), E(f(\tilde{X}_\alpha^+))]$. On the other hand, for non-increasing f, the α-cut is given by $\tilde{E}(f(\tilde{X}))_\alpha = [E(f(\tilde{X}_\alpha^+)), E(f(\tilde{X}_\alpha^-))]$. These extensions are also well-defined and have monotone and continuous properties in Proposition 2. We note that call options and put options can be represented by a continuous non-increasing function $f(x) = (\kappa - x)^+$ and a non-decreasing function $f(x) = (x - \kappa)^+$, and this kinds of problems are discussed by Yoshida [14], Kurano et al. [6] and so on.

3.6 Moments

Now we discuss the moments in statistics as an application of Theorem 1. Let P be a non-atomic probability on a sample Ω. Let $\mathcal{X}$ be the set of all integrable real random variables on (Ω, P). In statistics, for $n = 1, 2, \cdots$, the nth moment of an integrable real random variable $X(\in \mathcal{X})$ at a point $c(\in \mathbb{R})$ is given by $E((X - c)^n)$. The *nth moment* of an integrable fuzzy random variable $\tilde{X}(\in \tilde{\mathcal{X}})$ at $c(\in \mathbb{R})$ is given by the following proposition.

Proposition 4. *Let $n = 1, 2, \cdots$. The nth moment $\tilde{E}((\tilde{X} - c)^n)$ at $c(\in \mathbb{R})$ of an integrable fuzzy random variable $\tilde{X}$ is a fuzzy number whose α-cuts are given by*

$$\tilde{E}((\tilde{X} - c)^n)_\alpha = \left[E((\mathrm{med}\{\tilde{X}_\alpha^-, \tilde{X}_\alpha^+, c\} - c)^n), E(\max\{(\tilde{X}_\alpha^- - c)^n, (\tilde{X}_\alpha^+ - c)^n\})\right] \tag{21}$$

for even n, and

$$\tilde{E}((\tilde{X} - c)^n)_\alpha = \left[E((\tilde{X}_\alpha^- - c)^n), E((\tilde{X}_\alpha^+ - c)^n)\right] \tag{22}$$

for odd n, where $\mathrm{med}\{c_1, c_2, c_3\}$ is the median of three real numbers c_1, c_2, c_3.

Proof. In case where n is odd, since the nth moment function $f(x) = (x - c)^n$ is increasing and continuous, from Theorem 1 we obtain (22). Next, in case where n is even, we cannot apply Theorem 1 since the nth moment does not satisfy the monotone properties $(\varphi, \leq)$ and $(\varphi, \geq)$. Hence the α-cut of the nth moment for a fuzzy random variable $\tilde{X}$

$$\tilde{E}((\tilde{X} - c)^n)(x) = \sup_{X \in \mathcal{X} : E((X-c)^n)=x} \inf_{\omega \in \Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R}, \tag{23}$$

is given by

$$\tilde{E}((\tilde{X}-c)^n)_\alpha = \{E((X-c)^n) | X \in \mathcal{X},\, X(\omega) \in \tilde{X}_\alpha(\omega) \,\text{for all}\, \omega \in \Omega\}. \quad (24)$$

Therefore we investigate a set $\{(x-c)^n | \tilde{X}_\alpha^- \le x \le \tilde{X}_\alpha^+\}$. We have the lower bound

$$\min\{(x-c)^n | \tilde{X}_\alpha^- \le x \le \tilde{X}_\alpha^+\} = \begin{cases} (\tilde{X}_\alpha^+ - c)^n & \text{if } \tilde{X}_\alpha^+ < c \\ 0 & \text{if } \tilde{X}_\alpha^- \le c \le \tilde{X}_\alpha^+ \\ (\tilde{X}_\alpha^- - c)^n & \text{if } c < \tilde{X}_\alpha^-, \end{cases}$$

and this follows $\min\{(x-c)^n | \tilde{X}_\alpha^- \le x \le \tilde{X}_\alpha^+\} = (\text{med}\{\tilde{X}_\alpha^-, \tilde{X}_\alpha^+, c\} - c)^n$. On the other hand, we have the upper bound

$$\max\{(x-c)^n | \tilde{X}_\alpha^- \le x \le \tilde{X}_\alpha^+\} = \begin{cases} (\tilde{X}_\alpha^- - c)^n & \text{if } \tilde{X}_\alpha^+ < c \\ \max\{(\tilde{X}_\alpha^- - c)^n, (\tilde{X}_\alpha^+ - c)^n\} & \text{if } \tilde{X}_\alpha^- \le c \le \tilde{X}_\alpha^+ \\ (\tilde{X}_\alpha^+ - c)^n & \text{if } c < \tilde{X}_\alpha^-, \end{cases}$$

and this follows $\max\{(x-c)^n | \tilde{X}_\alpha^- \le x \le \tilde{X}_\alpha^+\} = \max\{(\tilde{X}_\alpha^- - c)^n, (\tilde{X}_\alpha^+ - c)^n\}$. Therefore, we obtain the interval

$$\begin{aligned} &\{(x-c)^n | \tilde{X}_\alpha^- \le x \le \tilde{X}_\alpha^+\} \\ &= \left[\text{med}\{\tilde{X}_\alpha^-, \tilde{X}_\alpha^+, c\} - c)^n, \max\{(\tilde{X}_\alpha^- - c)^n, (\tilde{X}_\alpha^+ - c)^n\}\right]. \end{aligned}$$

Thus we obtain (21) from (24) and this equation. □

3.7 Variances

Let P be a non-atomic probability on a set Ω. Let $\mathcal{X}$ be the set of all integrable real random variables on (Ω, P). The variance is given by

$$V(X) = E((X - E(X))^2) = \int (X - E(X))^2 \, dP. \quad (25)$$

The variance does not satisfy the monotone properties $(\varphi, \le)$ and $(\varphi, \ge)$. Hence we can give a definition of the extended estimation $\tilde{V}(\tilde{X})$ by

$$\tilde{V}(\tilde{X})(x) := \sup_{X \in \mathcal{X} : V(X) = x} \inf_{\omega \in \Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R}, \quad (26)$$

and this definition implies that its α-cut is

$$\{E((X - E(X))^2) | X \in \mathcal{X},\, X(\omega) \in \tilde{X}_\alpha(\omega) \,\text{for all}\, \omega \in \Omega\}. \quad (27)$$

In general, this set does not equal to the interval (6) since the monotone properties $(\varphi, \le)$ and $(\varphi, \ge)$ does not hold. Therefore we need to take another approach for this case (Refer to Conclusion). However, regarding the second moments at a point, in Section 4 we can propose a method which is based on an ideal from the previous section.

4 An Approach to General Cases

Finally we discuss a method for the cases where functionals φ do not satisfy the monotone properties $(\varphi, \leq)$ and $(\varphi, \geq)$. Let P be a non-atomic probability on a set Ω. Let $\mathcal{X}$ be the set of all integrable real random variables on (Ω, P), and let $\tilde{\mathcal{X}}$ be the set of all integrable fuzzy random variables on (Ω, P). Let $c(\in \mathbb{R})$ be a fixed constant. Let n be a positive even number, and put $f(x) := (x-c)^n$ for $x \in \mathbb{R}$. Then we have a representation

$$f(x) = \max\{f^+(x), -f^-(x)\} \tag{28}$$

for $x \in \mathbb{R}$, with two non-decreasing functions

$$f^+(x) := \begin{cases} 0 & \text{if } x < c \\ (\max\{x-c, 0\})^n & \text{if } x \geq c, \end{cases} \quad f^-(x) := \begin{cases} -(\min\{x-c, 0\})^n & \text{if } x < c \\ 0 & \text{if } x \geq c. \end{cases} \tag{29}$$

From Section 3.6, for the continuous non-decreasing functions $f^{\pm}$, the expected rewards of an integrable fuzzy random variable $\tilde{X}(\in \tilde{\mathcal{X}})$ are fuzzy numbers

$$\tilde{E}(f^{\pm}(\tilde{X}))(x) := \sup_{X \in \mathcal{X} : E(f^{\pm}(X)) = x} \inf_{\omega \in \Omega} \tilde{X}(\omega)(X(\omega)), \qquad x \in \mathbb{R}. \tag{30}$$

Therefore we obtain the following result.

Theorem 2. *The expectation $\tilde{E}(f(\tilde{X}))$ of the fuzzy random variable $f(\tilde{X})$ is a fuzzy number whose α-cuts are given by*

$$\begin{aligned} \tilde{E}(f(\tilde{X}))_\alpha &= \tilde{E}(\max\{f^+(\tilde{X}), -f^-(\tilde{X})\})_\alpha \\ &= \left[E(\max\{f^+(\tilde{X}_\alpha^-), -f^-(\tilde{X}_\alpha^+)\}), E(\max\{f^+(\tilde{X}_\alpha^+), -f^-(\tilde{X}_\alpha^-)\})\right] \end{aligned} \tag{31}$$

for $\alpha \in [0,1]$, and this term equals to (21) in case where $f(x) = (x-c)^n$ $(x \in \mathbb{R})$ for even n.

We note that the decomposition (28) of the function f is not unique, for example we can take $f = f^+ - f^-$. Then the result in Theorem 2 will be different. Therefore, we must take care of the choice of the decomposion of f.

5 Concluding Remarks

In this paper, we have discussed conditions and various properties of perception-based extension of estimations with randomness, and we have investigated several examples of the perception-based extension. These results are applicable to other estimations with fuzziness like linguistic data in engineering, economics, finance and so on.

We have also discussed an approach to cases where the monotone condition does not hold. Regarding the second moments at a point, we have proposed a method in Section 4. However, we cannot apply the results directly to variances

in Section 3.7. Actually, we have to take c is the expectation of a fuzzy random variable $\tilde{X}$ in (29) with $n = 2$. When c is a fuzzy number depending on the fuzzy random variable $\tilde{X}$, the analysis will be more difficult. We can find other approaches regarding variance in Carlsson and Fullér [1], Feng et al. [3], Körner [4] and Yoshida [15].

References

1. Carlsson, C., Fullér, R.: On Possibilistic Mean Value and Variance of Fuzzy Numbers. Fuzzy Sets and Systems 122, 315–326 (2001)
2. López-Díaz, M., Gil, M.A., Ralescu, D.A.: Overview on the Development of Fuzzy Random Variables. Fuzzy Sets and Systems 147, 2546–2557 (2006)
3. Feng, Y., Hu, L., Su, H.: The Variance and Covariance of Fuzzy Random Variables. Fuzzy Sets and Systems 120, 487–497 (2001)
4. Körner, R.: On the Variance of Fuzzy Random Variables. Fuzzy Sets and Systems 92, 83–93 (1997)
5. Kruse, R., Meyer, K.D.: Statistics with Vague Data. Riedel Publ. Co. Dortrecht (1987)
6. Kurano, M., Yasuda, M., Nakagami, J., Yoshida, Y.: A Fuzzy Stopping Problem with the Concept of Perception. Fuzzy Optimization and Decision Making 3, 367–374 (2004)
7. Kwakernaak, H.: Fuzzy Random Variables. Part I: Definitions and Theorem. Information Sciences 15, 1–29 (1978)
8. Kwakernaak, H.: Fuzzy Random Variables. Part II: Algorithms and Examples for the Discrete Case. Information Sciences 17, 253–278 (1979)
9. Meucci, A.: Risk and Asset Allocation. Springer, Heidelberg (2005)
10. Ohtsubo, Y.: Optimal Threshold Probability in Undiscounted Markov Decision Processes with a Target Set. Applied Math. and Computation 149, 519–532 (2004)
11. Puri, M.L., Ralescu, D.A.: Fuzzy Random Variables. J. Math. Anal. Appl. 114, 409–422 (1986)
12. Rockafellar, R.T., Uryasev, S.P.: Optimization of Conditional Value-at-Risk. Journal of Risk 2, 21–42 (2000)
13. Tasche, D.: Expected Shortfall and Beyond. Journal of Banking Finance 26, 1519–1533 (2002)
14. Yoshida, Y.: The Valuation of European Options in Uncertain Environment. European J. Oper. Res. 145, 221–229 (2003)
15. Yoshida, Y.: Mean Values, Measurement of Fuzziness and Variance of Fuzzy Random Variables for Fuzzy Optimization. In: Proceedings of SCIS & ISIS 2006 (Joint 3rd Intern. Conf. on Soft Comp. and Intell. Sys. and 7th Intern. Symp. on Advanced Intell. Sys.) Tokyo in Japan, pp. 2277–2282 (2006)
16. Zadeh, L.A.: Fuzzy Sets. Inform. and Control 8, 338–353 (1965)
17. Zadeh, L.A.: Toward a Perception-Based Theory of Probabilistic Reasoning with Imprecise Probabilities. J. of Statistical Planning and Inference 105, 233–264 (2002)

Boltzmann Machine Incorporated Hybrid Neural Fuzzy System for Hardware/Software Partitioning in Embedded System Design

Yue Huang and Yong-Soo Kim*

Department of Computer Science in Kyungwon University,
Sujeong-Gu, Seongnam-Si, Gyeonggi-Do, 461-701, Korea
huangyue@ku.kyungwon.ac.kr, kimys@kyungwon.ac.kr

Abstract. Nowadays one of the most vital problems in embedded system co-design is Hardware/Software (HW/SW) partitioning. Due to roughly assumed parameters in design specification and imprecise benchmarks for judging the solution's quality, embedded system designers have been working on finding a more efficient method for HW/SW partitioning for years. We propose an application of a hybrid neural fuzzy system incorporating Boltzmann machine to the HW/SW partitioning problem. Its architecture and performance estimation against other popular algorithm are evaluated. The simulation result shows the proposed system outperforms other algorithm both in cost and performance.

Keywords: Boltzmann machine, hybrid neural fuzzy system, hardware/ software partitioning, embedded system.

1 Introduction

1.1 HW/SW Partitioning

Since the emergence of desktop computer, perhaps one of the most important technologies that enabled ubiquitous computing is the embedded system. An embedded system is a special-purpose system combined of computer hardware and software. It is widely used in everyday life, such as in cellular telephones, automobiles, multimedia devices, automated supermarket stockers, etc., making our work and life more efficient and comfortable. In the past, the design of embedded system was carried on serially. Software development just followed after hardware development. But as the scale and complexity of the system increase, this way of design is not suitable any more, and a new method called HW/SW co-design was introduced. HW/SW co-design allows simultaneous design of a system's hardware and software components, thus exploiting the advantage of both, optimizing performance and cost, and allowing easier redesign when needed [1]. Different from the traditional method, HW/SW co-design focuses more on the cooperation of both hardware and software designers.

* Corresponding author.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 307–317, 2007.

Based on the granularity of system function analysis, current HW/SW co-design methodology is divided into two subdisciplines: one is based on coarse-grain task, called HW/SW co-synthesis, which uses task graph to describe the system function; another is based on fine-grain basic scheduling block (BSB), called HW/SW partitioning, which uses control data flow graph (CDFG) to describe the system function. In this paper, we mainly focus on the method for HW/SW partitioning.

The purpose of HW/SW partitioning methodology is to generate a standard way of implementing modules in hardware and software. As the IT and VLSI technology develop, both hardware and software becomes more complex and the boundary between them isn't as obvious as before. On one hand it is not surprising to see that some complicated arithmetic can be implemented in hardware; on the other hand it is also quite common to see that software in RISC can implement a function that used to be done in hardware. For the limited system size and development cycle, we urgently need a methodology that guides us to precise partitioning for the agile development and best performance.

1.2 Introduction About Hybrid Neural Fuzzy System

The key to fuzzy logic is to map an input space to an output space under the control of a list of if-then rules. These rules coming from the domain experts are linguistic and inaccurate, so a certain membership function is defined to translate rules into fuzzy set theory. The shape of the membership function has a strong impact on the final inference result. Though it is vital to the quality of fuzzy inference, no one knows which function is most suitable for solving the problem. Bias always exists, more or less. The combination of neural network and fuzzy logic can use the learning ability of neural network to enhance the accuracy of fuzzy inference. On the other hand, although a neural network is able to learn from given data, the trained neural network is generally regarded as a black box. Due to the randomly chosen weights, the final result can be released after at least hundreds of times of training. But fuzzy logic can be used to endow the nodes in neural network with specific meaning and initialize the weights with expert knowledge. By this means, the training time can be obviously reduced. In a word, the combination of fuzzy logic and neural network takes the merits of both and avoids their drawbacks [2]. The interest in hybrid neural fuzzy system has grown rapidly since it came out. Researchers have applied it to data classification, image retrieval, controlling, decision making and many other fields. As time goes on, its potential applicability will be found in more domains.

There are several approaches to the combination of neural network and fuzzy logic, such as fuzzy neural network, concurrent neuro-fuzzy model, cooperative neuro-fuzzy model, and hybrid neuro-fuzzy model [3]. We introduce a new algorithm based on hybrid neuro-fuzzy model and show superiority of the algorithm. In this case, the system may be either interpreted as a special neural network with fuzzy parameters, or as a fuzzy system implemented in a parallel distributed form [3] [4].

2 HW/SW Partitioning Using Hybrid Neural Fuzzy System

In embedded system, some modules implemented by hardware may have better effect than done by software and in other cases modules implemented by software may be

better than done by hardware. The "better" means the implementation needs less area consumption and less executing time compared to the other way under the coordinate condition. For instance, if bit operation can be better implemented in hardware than done in software, it is done in hardware; and if file I/O is hard to implement in hardware, it is better to be done in software.

People have been doing research on HW/SW partitioning since mid 1990s. Many papers have been published in this domain with the methods of using ILP(integer liner programming)/MILP(mixed integer linear programming) algorithm, dynamic programming algorithm, genetic algorithm, simulated annealing algorithm, soft computing methodology, etc. In this paper, we propose a new idea for HW/SW partitioning using a hybrid neural fuzzy system. In section 2.1 some basic concepts and mathematical formulae will be mentioned. In section 2.2 the architecture of the hybrid neural fuzzy system will be introduced. In section 2.3 the simulation result will be presented.

2.1 Specification of the HW/SW Partitioning Problem

An embedded system can be expressed as a Directed Acyclic Graph (DAG). We take the specification introduced in paper [5], [6], [7] and [8] as reference. In the graph, node represents module and edge indicates some kind of communication between two modules. The DAG can be represented in form of:

$$G = (N, E), \tag{1}$$

where N is the node set, $N=\{n_0,n_1,\ldots\ldots n_{k-1}\}$. N is composed of two parts, $N=N_s+ N_h$. N_s is the subset of the nodes that are implemented in software and N_h is the subset of the nodes that are implemented in hardware, $|N_s|+|N_h| = k$, $N_s \cap N_h=\Phi$; E is the edge set, $E=\{e_{ij}\}$, $0 \leq i, j \leq k-1$, $E \subseteq N \times N$.

For each node $n_i \in N$,

$$n_i = (ah_i, th_i, as_i, ts_i, tm_i, F_i), \tag{2}$$

where ah_i indicates the area consumption for module *i* when this function is implemented in hardware, which may be represented by the number of FPGA modules;

th_i is the time consumption when the function is implemented in hardware;
as_i is the area consumption when the function is implemented in software, which may be represented by the amount of memory required for the program logic;
ts_i is the time consumption when this function is implemented in software;
tm_i is the number of times the module *i* has executed;
F_i is a binary mapping function, where $F_i=1$ means node n_i should be mapped into hardware set N_h and $F_i=0$ means node n_i should be mapped into software set N_s. For each node, its value is 0 initially.

In the rest of this paper, we will call these elements in formula (2) as main properties of module *i*.

For each edge $e_{ij} \in E$,

$$e_{ij} = (wt_{ij}), \tag{3}$$

$$wt_{ij} = t_{sisj} \times (1-F_i) \times (1-F_j) + t_{sihj} \times (1-F_i) \times F_j + t_{hisj} \times F_i \times (1-F_j) + t_{hihj} \times F_i \times F_j \tag{4}$$

where wt_{ij} indicates the input communication of node n_j from node n_i, or output communication of node n_i to node n_j, which presents the time spent on transferring data between output and input module. We define that wt_{ij} is composed of the following items:

t_{sisj} is the time consumption on data transfer when module *i* and *j* are implemented in software;
t_{sihj} is the time consumption on data transfer when module *i* is implemented in software and module *j* is implemented in hardware;
t_{hisj} is the time consumption on data transfer when module *i* is implemented in hardware and module *j* is implemented in software;
t_{hihj} is the time consumption on data transfer when module *i* and *j* are implemented in hardware.

We suppose some preparatory works have been done before our research begins, such as doing HW/SW specification and verification, estimating ah_i and th_i using ASAP (As-Soon-As-Possible) scheduling algorithm, estimating as_i and ts_i though DFG liner technology and selection of different micro-process, etc. Our algorithm has to be applied after these important parameters have been calculated. We can use them directly to infer the value F_i for each module. HW/SW partitioning is a NP-Hard problem. If there are *k* nodes in the DAG, there will be 2^k ways to do HW/SW partitioning. What we have to do is finding out a way from them with the least time consumption (TC) and the least area consumption (AC) under the time cost constraint TC_{con} and the area cost constraint AC_{con}. Based on the study in [5] and [7], we modified the formulae which are used to denote the partitioning problem as in formula (5) to (9):

$$\begin{aligned} TC = ST - \sum_{i=0}^{k-1} tm_i \times (ts_i - th_i) \times F_i + \\ \sum_{i=0}^{k-1} \sum_{j=0, j\neq i}^{k-1} tm_i \times \left[t_{sisj} \times (1-F_i) \times (1-F_j) + t_{sihj} \times (1-F_i) \times F_j + t_{hisj} \times F_i \times (1-F_j) + t_{hihj} \times F_i \times F_j \right] \end{aligned} \tag{5}$$

$$ST = \sum_{i=0}^{k-1} ts_i \tag{6}$$

$$AC = \sum_{i=0}^{k-1} ah_i \times F_i + \sum_{i=0}^{k-1} as_i \times (1-F_i) \times \alpha \tag{7}$$

$$TC \leq TC_{con} \tag{8}$$

$$AC \leq AC_{con} \tag{9}$$

Hitherto, area consumption of a software module has not been taken into the calculation of the system's area consumption. But we regard as_i a valuable factor in calculating AC and included it in formula (7). Size becomes one of the most important parameters when we evaluate an embedded system. To decrease the size, system designer always tries to use as less registers as possible. If a software module uses a lot of registers which are useless or unnecessary for other modules, we have to reconsider its implementation mode carefully. Based on this consideration, we redefine formula (7) and take as_i with a coefficient α in it. α is correlative with the importance of the number of registers and the size of the memory. Heuristically, $\alpha \in [0.1, 0.5]$. Finally, the cost function can be defined as:

$$\text{cost} = \frac{\text{AC}_{\text{con}} - \text{AC}}{\text{AC}_{\text{con}}} + \frac{\text{TC}_{\text{con}} - \text{TC}}{\text{TC}_{\text{con}}} \tag{10}$$

2.2 Structure of the Hybrid Neural Fuzzy System

Our proposed hybrid neural fuzzy system is composed of two parts: a classification network and an operation network. The classification network is used to generate an initial scheme for the HW/SW partitioning problem by using fuzzy logic. Its outputs are transferred into the operation network to generate the optimal partitioning scheme under the constraints. In an overview, the whole system is working as a fuzzy system, while in terms of its topological structure, it is a network composed of nodes and weights. Therefore it combines the advantages of both fuzzy system and neural network with a great mapping and self-learning ability.

2.2.1 The Structure of the Classification Network

We propose five layers in the classification network. Neurons in the input layer are assumed to be the nodes in the DAG. The input data are the vectors composed of elements in formula (2), which indicate the main properties of each node in the DAG. Three layers between input and output layers are hidden in the black box.

The first hidden layer is called fuzzy layer. A membership function is used to classify the vector element into several classes, which are the fuzzy sets. Supposing there are c elements in each input vector and each vector element data can be classified into s classes, there are $c \times s$ neurons in this layer in total.

The second hidden layer is called rule layer. The outputs of the fuzzy layer are transferred into this layer to match the rules from domain experts. If there are r rules, there will be r neurons in this layer.

Because every rule neuron in the rule layer gives out a prediction and the predictions may be conflicting, we define the third hidden layer as a confirm layer to calculate the confidence of the final prediction, which is represent as in formula (11) and (12),

$$\text{confidence} = \begin{cases} \dfrac{\text{a}}{\text{r}} & \text{if } \text{a} > \dfrac{\text{r}}{2} \\ 1 - \dfrac{\text{a}}{\text{r}} & \text{if } 0 \le \text{a} \le \dfrac{\text{r}}{2} \end{cases} \tag{11}$$

$$a=\sum_{i=0}^{r-1} 1\times O_{rulelayer\text{-}i} \tag{12}$$

where r is the number of neurons in rule layer; $O_{rulelayer\text{-}i}$ is the output of neuron i in rule layer, with the binary value 0 or 1.

The last layer is output layer. Each input neuron has a corresponding output neuron to show the final predication for the input vector. As shown in formula (13), O_i with value 0 indicates input neuron should be implemented in software and with value 1 indicates input neuron should be implemented in hardware.

$$O_i=\begin{cases} 1 & \text{if } a > \dfrac{r}{2} \\ 0 & \text{if } 0 \le a \le \dfrac{r}{2} \end{cases} \tag{13}$$

The structure of the classification network is shown in Figure 1. To make its architecture clear and easy to see, we only show the connections for one input neuron n_i.

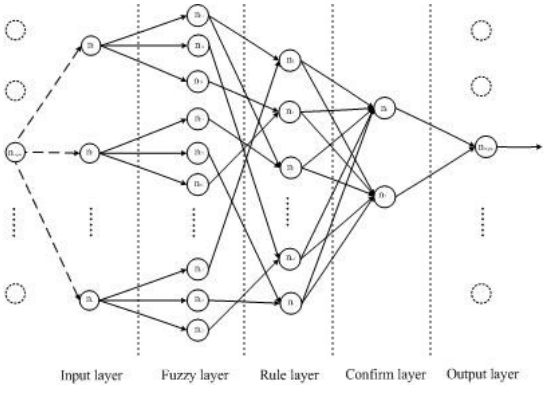


Fig. 1. Structure of the classification network

After running this classification network, we can get an initial partitioning scheme: every node is marked by an alternative number 0 or 1, indicating the implementation mode.

2.2.2 The Structure of the Operation Network

The operation network is a neural network with Boltzmann machine. Boltzmann machine is put forward to solve the local minimum problem of both BP neural network (which means the MLP trained via BP training algorithm) and Hopfield neural network [9] [10], so it combines the advantage of their network structure, learning algorithm and dynamic mechanism. Similar to BP neural network, Boltzmann machine also has layers, called visual layer and hidden layer. And visual layer can be divided into input part and output part. The differences between Boltzmann machine and other multiple layer networks are that the layers in Boltzmann machine have no obvious circumscription and the connection between two neurons is bidirectional. Like Hopfield neural network, all the neurons in the network

are connected and the change of network state obeys the probability distribution. This combination helps Boltzmann machine to avoid falling into the local minimum of error function or energy function and finally get the optimal solution of the problem. Following Figure 2 shows the network structure of a simple Boltzmann machine [11].

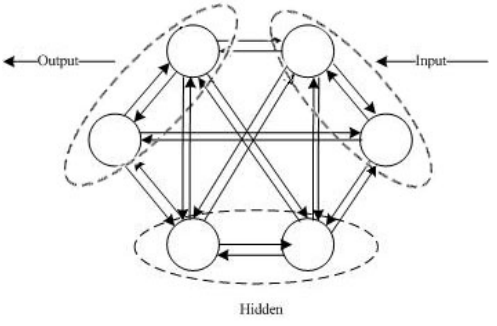


Fig. 2. Network structure of a simple Boltzmann machine

Boltzmann machine is well known for solving NP problem. In our problem, the input data are the character properties which are the same as classification network's. The output is a sequence composed of binary number 0 or 1. Initial partitioning scheme gotten from the classification network is the initial state of the operation network. Formula (10) is working as the energy function. If F_i=1, ah_i, th_i, t_{hisj} and t_{hihj} are enabled and taken into the calculation of *cost*; if F_i =0, as_i, ts_i, t_{sisj} and t_{sihj} are enabled and considered. For every iteration, the network tries to get a balanced state for all the neurons under the controlling parameter *T*. As iteration goes on, both parameter *T* and energy *cost* decreases. When *T* becomes small enough and the network reaches a balanced state, the network's output is the optimized solution for the problem.

2.3 Performance Evaluation by Simulation

The theory behind our system indicates that if the Boltzmann machine receives more crisp meaningful inputs, it will improve the overall output and quality of its prediction. To verify the availability of our proposed system, we adopt the similar simulation method introduced in [12] and [13]. For there is no standard test data in this domain, we generate a few DAGs randomly and arrange the attributes to each node and edge. We assume these are the practical data gotten from previous work, such as system performance analysis.

2.3.1 Simulation System Architecture

During our study on designing an embedded system, we always keep a viewpoint in mind that embedded system is still a kind of computer system but the processor's architecture and its software are a little different from the typical conventional computer. Therefore for simulation purpose, we assume the system model has ten basic modules like a real embedded system typically has, shown in Table 1. Some of

the modules are HW-bound, some are SW-bound, and some are mixed. Nodes in the DAG are distributed into these modules. To make the simulation more reflective of the real system, we allocate the properties for each node based on its module.

Table 1. System modules

HW-bound	HW-SW	SW-bound
CPU	Net	File System
RAM	Driver	Scheduling
Flash Memory	Loader	Inter-Processor Call
	Memory Management	

2.3.2 Simulation Data, Fuzzy Rules and the Neural Network

For simulation, we generate six DAGs randomly by using GVF (Graph Visualization Framework), with 50, 100, 200, 400, 800, 1500 nodes respectively. Each edge and node in the graph has different character properties. To enlarge the simulation data base, we add the Gaussian noise at 10 different levels to the 6 groups of sample data. As a result, there are 30500 sample data for testing in total. Figure 3 shows a graph generated by GVF with 100 nodes and 120 edges.

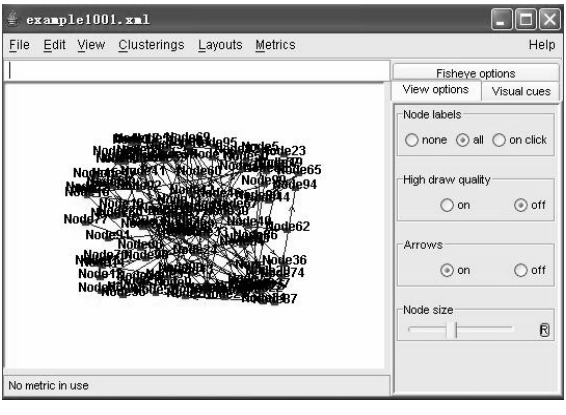


Fig. 3. An example graph with 100 nodes and 120 edges

Since currently there are no standard fuzzy rules from the domain experts available for use, we choose 200 sample data from the simulation data base as the learning data. An open-source data mining software WEKA [15] is used to get the fuzzy rules for our hybrid neural fuzzy system. WEKA can work on the learning data and generate fuzzy rules automatically.

The input data are the properties vectors fed into the hybrid system; and the partitioning scheme, which is a 0-1 sequence, is the output data. The main logic in classification network and operation network is implemented using MATLAB. In classification network, we choose sigmoid membership function to classify original data into fuzzy sets. We take the coefficient α in formula (7) as α =0.2, and the

controlling parameter T in Boltzmann machine as T_0=200 (the initial value of T), T_{final} =0.0001 (the final value of T). According to the simulation results with different coefficients, we found when α is more than 0.5, sometimes AC couldn't be under the constraint AC_{con}. And also, when α is less than 0.1, as_i becomes too small to be meaningful. That's the reason why we recommend the value of α should be between 0.1 and 0.5 and we choose α=0.2 in our simulation.

2.3.3 Simulation Results

The following Table 2 shows the comparison of our proposed hybrid neural fuzzy system (HNFS) with simulated annealing algorithm (SAA). The value of AC, TC and *cost* are listed. According to the data, AC shows no obvious difference between these two methods, but TC calculated by HNFS is closer to TC_{con} than the one calculated by SAA. So the *cost* of HNFS is less than that of SAA.

Table 2. Performance comparison of HNFS with SAA

	AC_{con}	TC_{con}	HNFS			SAA		
			AC	TC	cost	AC	TC	cost
50	8866	434380	8757	389550	0.1155	8751.8	377860	0.1430
100	17384	1268400	17366	1192540	0.1608	17360	1030058	0.1892
200	34626	2526800	34622	2100190	0.1689	34613	1937770	0.2332
400	64263	4683800	64238	4017290	0.1423	64242	3846782	0.1787
800	128558	10969000	128551	9121000	0.1684	128555	8519388	0.1959
1500	235574	19469000	235574	17013280	0.1262	235572	16012500	0.1775

The Figure 4 is the running time of these two methods. The curve shows the time HNFS spends in getting the final result is obviously less than that of SAA. Because the initial state of HNFS is trained by fuzzy logic, it is more meaningful than SAA's random initial state.

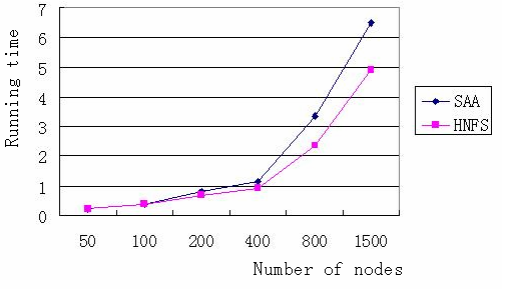


Fig. 4. Running time curve of HNFS and SAA

3 Conclusion

In this paper, we introduce a method of applying hybrid neural fuzzy system into the HW/SW partitioning problem. The main advantage of the hybrid neural fuzzy system

is accomplishing the task with high accuracy and less running time. Firstly, we use a simple neural network with fuzzy logic mechanism to match the character properties to the expert rules and generate an initial partitioning scheme. Then the scheme is transferred into a neural network based on Boltzmann machine. As iteration goes on, partitioning scheme is modified. When the network reaches a balanced state and the control parameter becomes small enough, the output of the operation network is regarded as the final scheme for HW/SW partitioning problem. The simulation result has demonstrated our method for HW/SW partitioning is viable and has a better performance than some of the current methods in this research domain.

Acknowledgement

This research was supported by the Ministry of Education, Seoul, Korea, under the BK21 project.

References

1. Catania, V., Malgeri, M., Russo, M.: Applying Fuzzy Logic To Codesign Partitioning. IEEE Micro archive 17(3), 62–70 (1997)
2. Nauck, D., Klawonn, F., Kruse, R.: Combining Neural Network and Fuzzy Controllers. In: Klement, E.P., Slany, W. (eds.) FLAI 1993. LNCS, vol. 695, Springer, Heidelberg (1993)
3. Nauck, D.: Learning Methods for Fuzzy Systems. In: proceedings of the Third European Congress on Intelligent Techniques and Soft Computing EUFIT'95, Aachen, August 28-31,1995, pp. 1159–1164 (1995)
4. Nauck, D., Kruse, R.: Choosing Appropriate Neuro-Fuzzy Models. In: Proc. EUFIT'94, Aachen, pp. 552–557 (1994)
5. Ma, T.-Y., Li, Z.-Q., Yang, J.: A Novel Neural Network Search for Energy-Efficient Hardware-Software Partitioning. In: Machine Learning and Cybernetics, 2006. International Conference on August 2006, pp. 3053–3058 (2006)
6. Arato, P., Juhasz, S., Mann, Z.A, Orban, A., Papp, D.: Hardware-software partitioning in embedded system design. In: Intelligent Signal Processing, 2003. IEEE International Symposium on, September 4-6, 2003, pp. 197–202. IEEE Computer Society Press, Los Alamitos (2003)
7. Mitra, R.S., Basu, A.: Hardware-software partitioning: a case for constraint satisfaction. Intelligent Systems and Their Applications 15(1), 54–63 (2000)
8. Ma, T., Wang, X., Li, Z.: Neural Network Optimization for Hardware-Software Partitioning. Innovative Computing, Information and Control, vol. 3, pp. 423–426 (2006)
9. Lin, C.T., Lee, C.S.G.: A multi-valued Boltzman machine. Systems Man and Cybernetics, IEEE Transaction on 25(4), 660–669 (1995)
10. Ma, H.: Pattern Recognition using Boltzmann machine. In: Southeastcon '95, Visualize the Future, Proceedings IEEE, pp. 23–29 (March 26-29, 1995)
11. Fecit science and technology development center: Neural network theory and MATLAB7 application. By Publishing House of Electronics Industry, p. 158
12. Xiong, Z., Chen, J., Li, S.: Hardware/Software partitioning for platform-based design method. In: Design Automation Conference, 2005. Proceeding of the ASP-DAC 2005. Asia and South Pacific, January 18-21, 2005, vol. 2, pp. 691–696 (2005)

13. Wang, G., Gong, W.R., Kastner, R.: A New Approach for Task Level Computational Resource Bi-partitionging. In: Proc. of the IASTED Int'l Conf. on Parallel and Distributed Computing and Systems (PDCS), pp. 434–444. ACTA Press (2003)
14. Mudry, P.A., Zufferey, G., Tempesti, G.: A hybrid genetic algorithm for constrained hardware-software partitioning. In: Design and Diagnostics of Electronic Circuits and system, 2006 IEEE, April 18-21, 2006, pp. 1–6. IEEE Computer Society Press, Los Alamitos (2006)
15. http://www.cs.waikato.ac.nz/ml/weka

Artificial Bee Colony (ABC) Optimization Algorithm for Training Feed-Forward Neural Networks

Dervis Karaboga, Bahriye Akay, and Celal Ozturk

Erciyes University, Engineering Faculty, Department of Computer Engineering
karaboga@erciyes.edu.tr,
bahriye@erciyes.edu.tr, celal@erciyes.edu.tr

Abstract. Training an artificial neural network is an optimization task since it is desired to find optimal weight set of a neural network in training process. Traditional training algorithms has some drawbacks such as getting stuck in local minima and computational complexity. Therefore, evolutionary algorithms are employed to train neural networks to overcome these issues. In this work, Artificial Bee Colony (ABC) Algorithm which has good exploration and exploitation capabilities in searching optimal weight set is used in training neural networks.

1 Introduction

Since Artificial Neural Networks (ANNs) are quite successful in modelling nonlinearity and have characteristics such as being capable of generalizing, adaptability, self-organizing, real time operation and fault tolerance, they are involved in so many applications in research fields. Finding a suitable network structure and finding optimal weight values make design of ANNs difficult optimization problems. In other words, the success of ANNs largely depends on the architecture, the training algorithm, and the choice of features used in training.

Artificial neural network training has traditionally been carried out using the back-propagation (BP) gradient descent (GD) algorithm [1]. But this technique has some drawbacks such as dependence of error surface shape, initial values of connection weights, parameters. If the error surface is multimodal, the gradient descent based algorithms are trapped at local minima. Involving differentiation of error function is another issue with this kind of algorithms [2]. Saturation may occur if the output is pushed towards its extremes at some point before convergence is reached, so that the derivative is too small to make further significant weight changes, causing the network to settle in an incorrect local minimum or reach a state of network paralysis . This saturation may occur for a number of reasons, most of which are easily avoided [3].In order to overcome the disadvantages of gradient based algorithms, many global optimization methods have been proposed for training feed-forward neural networks such as Genetic Algorithms [4,5,6,7,8,9,10,11,12,13,14,15,16,17], The Particle Swarm Optimization algorithm [18,19,20,22,21,23], Differential Evolution [24,25,26,27,28,29] and

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 318–329, 2007.

Evolutionary Programming algorithms [30,31,32]. Also, some hybrid techniques combining traditional techniques such as back propagation and evolutionary algorithms are proposed for training neural networks [33]. Not all of this algorithms handle with only connection weights, they also optimize the structure of the network. However, when the neural network training becomes a large scale, the number of network parameters grows drastically. For example,learning a huge number of hidden layer weights in a multi-layer perceptron (MLP) neural network can be considered as a large scale optimization problem.

Karaboga has described Artificial Bee Colony (ABC) algorithm based on the foraging behaviour of honey bees for numerical optimization problems [34], and Karaboga and Basturk have compared the performance of the ABC algorithm with those of other well-known modern heuristic algorithms such as Genetic Algorithm (GA), Differential Evolution (DE), Particle Swarm Optimization (PSO) on unconstrained problems [35]. In this work, the ABC algorithm is employed in training feed-forward neural networks and the performance of the algorithm is compared with Genetic Algorithm (GA) from evolutionary algorithms and Back-Propagation (BP) Algorithm. The paper is organized as follows: In Section 2, training an artificial neural network is described. In Section 3, implementation of artificial neural network training by using the ABC algorithm is introduced. In Section 4, experiments and results of produced by ABC, GA, and BP are presented and discussed.

2 Training Feed Forward Artificial Neural Networks

An ANN consists of a set of processing elements (Fig. 1), also known as neurons or nodes, which are interconnected with each other [14]. Output of the ith neuron can be described by Eq. 1

$$y_i = f_i(\sum_{j=1}^{n} w_{ij}x_j + \theta_i) \tag{1}$$

where y_i is the output of the node, x_j is the jth input to the node, w_{ij} is the connection weight between the node and input x_j, θ_i is the threshold (or bias) of the node, and f_i is the node transfer function. Usually, the node transfer function is a nonlinear function such as a heaviside function, a sigmoid function, a Gaussian function, etc.

Generally, the adaptation can be carried out by minimizing (optimizing) the network error function E. The error function is given by Eq. 2:

$$E(\boldsymbol{w}(t)) = \frac{1}{n}\sum_{j=1}^{n}\sum_{k=1}^{K}(d_k - o_k)^2 \tag{2}$$

where, $E(\boldsymbol{w}(t))$ is the error at the tth iteration; $\boldsymbol{w}(t)$, the weights in the connections at the tth iteration; d_k, the desired output node; o_k, the actual value of the kth output node;K, the number of output nodes; n, the number of patterns.

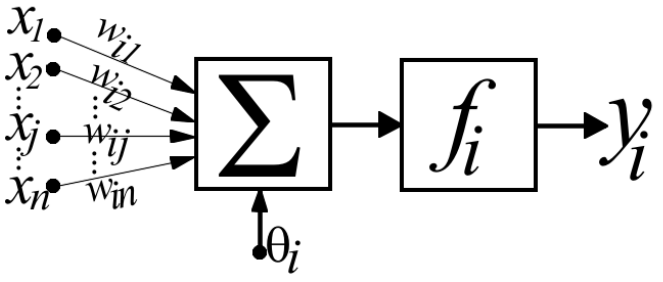


Fig. 1. Processing unit of an ANN (neuron)

The optimization goal is to minimize the objective function by optimizing the network weights $\boldsymbol{w}(t)$. In Evolutionary Algorithms, the major idea underlying this synthesis is to interpret the weight matrices of the ANNs as individuals, to change the weights by means of some operations such as crossover and mutation, and to use the error E produced by the ANNs as the fitness measure which guides selection. This leads to the following evolutionary training cycle [36]:

1. Formation of the next population of ANNs by means of operators such as crossover and mutation and fitness–oriented selection of the the weight matrices. (The initial population is randomly created.)
2. Evaluation of the fitness values of the ANNs.
3. If the desired result is obtained, then stop; otherwise goto step 1.

3 Artificial Bee Colony Algorithm

Artificial Bee Colony (ABC) algorithm was proposed by Karaboga for optimizing numerical problems in 2005 [34]. The algorithm simulates the intelligent foraging behaviour of honey bee swarms. It is a very simple, robust and population based stochastic optimization algorithm. Karaboga and Basturk have compared the performance of the ABC algorithm with those of other well-known modern heuristic algorithms such as Genetic Algorithm (GA), Differential Evolution (DE), Particle Swarm Optimization (PSO) on unconstrained problems [35].

Detailed pseudo-code of the ABC algorithm is given below:

1: Initialize the population of solutions $x_i, i = 1 \ldots SN$
2: Evaluate the population
3: cycle=1
4: **repeat**
5: Produce new solutions v_i for the employed bees by using (4) and evaluate them
6: Apply the greedy selection process
7: Calculate the probability values p_i for the solutions x_i by (3)
8: Produce the new solutions v_i for the onlookers from the solutions x_i selected depending on p_i and evaluate them

9: Apply the greedy selection process
10: Determine the abandoned solution for the scout, if exists, and replace it with a new randomly produced solution x_i by (5)
11: Memorize the best solution achieved so far
12: cycle=cycle+1
13: **until** cycle=MCN

In ABC algorithm, the position of a food source represents a possible solution to the optimization problem and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution. The number of the employed bees or the onlooker bees is equal to the number of solutions in the population. At the first step, the ABC generates a randomly distributed initial population $P(G = 0)$ of SN solutions (food source positions), where SN denotes the size of population. Each solution x_i $(i = 1, 2, ..., SN)$ is a D-dimensional vector. Here, D is the number of optimization parameters. After initialization, the population of the positions (solutions) is subjected to repeated cycles, $C = 1, 2, ..., MCN$, of the search processes of the employed bees, the onlooker bees and scout bees. An employed bee produces a modification on the position (solution) in her memory depending on the local information (visual information) and tests the nectar amount (fitness value) of the new source (new solution). Provided that the nectar amount of the new one is higher than that of the previous one, the bee memorizes the new position and forgets the old one. Otherwise she keeps the position of the previous one in her memory. After all employed bees complete the search process, they share the nectar information of the food sources and their position information with the onlooker bees on the dance area. An onlooker bee evaluates the nectar information taken from all employed bees and chooses a food source with a probability related to its nectar amount. As in the case of the employed bee, she produces a modification on the position in her memory and checks the nectar amount of the candidate source. Providing that its nectar is higher than that of the previous one, the bee memorizes the new position and forgets the old one.

An artificial onlooker bee chooses a food source depending on the probability value associated with that food source, p_i, calculated by the following expression (3):

$$p_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n} \tag{3}$$

where fit_i is the fitness value of the solution i which is proportional to the nectar amount of the food source in the position i and SN is the number of food sources which is equal to the number of employed bees (BN).

In order to produce a candidate food position from the old one in memory, the ABC uses the following expression (4):

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \tag{4}$$

where $k \in \{1, 2,..., SN\}$ and $j \in \{1, 2,..., D\}$ are randomly chosen indexes. Although k is determined randomly, it has to be different from i. $\phi_{i,j}$ is a random

number between [-1, 1]. It controls the production of neighbour food sources around $x_{i,j}$ and represents the comparison of two food positions visually by a bee. As can be seen from (4), as the difference between the parameters of the $x_{i,j}$ and $x_{k,j}$ decreases, the perturbation on the position $x_{i,j}$ gets decrease, too. Thus, as the search approaches to the optimum solution in the search space, the step length is adaptively reduced.

The food source of which the nectar is abandoned by the bees is replaced with a new food source by the scouts. In ABC, this is simulated by producing a position randomly and replacing it with the abandoned one. In ABC, providing that a position can not be improved further through a predetermined number of cycles, then that food source is assumed to be abandoned. The value of predetermined number of cycles is an important control parameter of the ABC algorithm, which is called "*limit*" for abandonment. Assume that the abandoned source is x_i and $j \in \{1, 2,..., D\}$, then the scout discovers a new food source to be replaced with x_i. This operation can be defined as in (5)

$$x_i^j = x_{\min}^j + \text{rand}(0, 1)(x_{\max}^j - x_{\min}^j) \tag{5}$$

After each candidate source position $v_{i,j}$ is produced and then evaluated by the artificial bee, its performance is compared with that of its old one. If the new food source has an equal or better nectar than the old source, it is replaced with the old one in the memory. Otherwise, the old one is retained in the memory. In other words, a greedy selection mechanism is employed as the selection operation between the old and the candidate one.There are three control parameters in the ABC: The number of food sources which is equal to the number of employed or onlooker bees (SN), the value of *limit*, the maximum cycle number (MCN).

In a robust search process, exploration and exploitation processes must be carried out together. In the ABC algorithm, while onlookers and employed bees carry out the exploitation process in the search space, the scouts control the exploration process.

4 Experimental Study

In this work, three problems are considered: XOR, 3-Bit Parity and 4-Bit Encoder-Decoder problems, which are benchmark problems used in training neural networks.

4.1 The Exclusive-OR Problem

The first test problem we used in the experiments is the exclusive-OR (XOR) Boolean function which is a difficult classification problem mapping two binary inputs to a single binary output as (0 0;0 1;1 0;1 1)→(0;1;1;0). In the simulations we use a 2-2-1 feed-forward neural network with six connection weights, no biasses (having six parameters, XOR6) and a 2-2-1 feed-forward neural network

with six connection weights and three biases (having 9 parameters, XOR9) and a 2-3-1 feed-forward neural network having nine connection weights and four biases totally thirteen parameters (XOR13). For XOR6, XOR9 and XOR13 problems, the parameter ranges [-100,100], [-10,10] and [-10,10] are used, respectively.

4.2 3-Bit Parity Problem

The second test problem is the three bit parity problem. The problem is taking the modulus 2 of summation of three inputs. In other words, if the number of binary inputs is odd, the output is 1, otherwise it is 0. (0 0 0;0 0 1;0 1 0;0 1 1; 1 0 0; 1 0 1; 1 1 0; 1 1 1)→(0;1;1;0;1;0;0;1) We use a 3-3-1 feed-forward neural network structure for the 3-Bit Parity problem. It has twelve connection weights and four biasses, totally sixteen parameters. The parameter range was [-10,10] for this problem.

4.3 4-Bit Encoder/Decoder Problem

The third problem is 4-bit encoder/decoder problem. The network is presented with 4 distinct input patterns, each having only one bit turned on. The output is a duplication of the inputs. (0 0 0 1;0 0 1 0;0 1 0 0;1 0 0 0)→(0 0 0 1;0 0 1 0;0 1 0 0;1 0 0 0). This is quite close to real world pattern classification tasks, where small changes in the input pattern cause small changes in the output pattern [37]. A 4-2-4 feed-forward neural network structure is used for this problem and it has totally 22 parameters including sixteen connection weights and six biases. For this problem, the parameter range is [-10,10].

The parameter ranges, dimension of the problems, and the network structures are presented in Table 1.

Table 1. Parameters of the problems considered in the experiments. D: Dimension of the problem.

	Range	NN Structure	D
XOR6	[-100,100]	2-2-1 without Bias	6
XOR9	[-10,10]	2-2-1 with bias	9
XOR13	[-10,10]	2-3-1 with bias	13
3-Bit Parity	[-10,10]	3-3-1 with bias	16
4-Bit Enc.-Dec.	[-10,10]	4-2-4 with bias	22

4.4 Settings

Experiments were repeated 30 times for each case and each run was started with a random population with different seeds. In each network selected for each problem, sigmoid function is employed as transfer function. Training processes were stopped when the mean squared error of the outputs associated with inputs was equal to or less than 0.01 (MSE$\leq$0.01) or when the maximum generation or cycle or epoch has been reached. Since the difficulty of each problem is different, different parameter settings were used for each of them. Among all of

the problems, XOR6 is the most difficult one since it does not employ biases. Therefore, all algorithms were run through more generations/cycles/epochs for this problem.

ABC Settings: The value of "*limit*" is equal to SN x D where D is the dimension of the problem. Colony size $(2 * SN)$ is 50 for all problems

GA Settings: In the experiments, roulette wheel selection scheme, single point crossover with the rate of 0.8, uniform mutation with the rate of 0.05 are employed. Generation gap is set to 0.9. The population size in GA were 50 for all problems.

BP Settings: In back-propagation experiments, NNs were trained by using Levenberg-Mardquart (LM) and Gradient Descent (GD) training algorithms. Learning rate for GD is 0.8.

These settings are summarized in Table 2.

Table 2. Values for control parameters of the algorithms. LR: Learning Rate, Pop: Population Size, CR: Crossover Rate, MR: Mutation Rate, GP: Generation Gap, SN: Colony Size.

BP	**GA**	**ABC**
LR: 0.8	Pop: 50	SN: 50
	CR: 0.8	limit: SN*D
	MR: 0.05	
	GP: 0.9	

Maximum cycle number (MCN) for ABC and maximum generation number for GA were 7500,100,75,1000 1000 for XOR6, XOR9, XOR13, 3-Bit Parity and 4-Bit Encoder-Decoder problems, respectively. Hence, the total objective function evaluation numbers were 375 000, 5000, 3750, 50000 and 50000 for the problems, respectively. In case of BP algorithm, the number of epochs for problems were 32000, 500, 250, 1600 and 2100, respectively. These values are presented in Table 3.

Table 3. Maximum Cycle/Generation/Epoch numbers for ABC/GA/BP and the total Objective Function Evaluation (OFE) numbers for algorithms

	Cycle/Gen.	**OFE**	**Epoch**
XOR6	7500	375000	32000
XOR9	100	5000	500
XOR13	75	3750	250
3-Bit Parity	1000	50000	1600
4-Bit Enc.-Dec.	1000	50000	2100

4.5 Results and Discussion

Statistical results and the success rates of the algorithms for XOR6, XOR9, XOR13, 3-Bit Parity and 4-Bit Encoder-Decoder problems are given in Tables 4-6. For XOR6 problem, back propagation algorithm trained by GD has 3 % success and back propagation algorithm trained by LM has 6% success. Genetic algorithm has 0% success while the ABC algorithm has 100 % success. On XOR6 problem, GA could not find the global minima. Mean cycle number of ABC algorithm is 2717.4 for XOR6 problem. Mean cycle number of ABC is more than the mean epoch number of the BP (LM), but the LM algorithm has got stuck with the local minima of XOR6 problem. For XOR9 problem, BP (GD) has 0% success while BP (LM) has 66.66 % success with the mean of 13 epochs. GA has 40 % success with a mean of 77.067 generations. The ABC algorithm has 100 % success and the mean of cycles is 32. As seen from the mean OFE and success numbers, the BP (LM) algorithm has very fast convergence speed but it gets stuck to the local minima while ABC algorithm goes on searching without being stuck in local minima. For XOR13 problem, BP (GD) has 0% success with an average of 250 epochs while BP (LM) has 96.66% success with an average of 9 epochs. On XOR13 problem, GA has 76.66 % success rate and ABC has 100 % success with an average of 28.2 cycles. Although the problems were the same in the case of XOR6, XOR9 and XOR13, the network structures employed in the experiments were different. The NN used for XOR13 problem has 3 hidden neurons while the networks employed for XOR6 and XOR9 have 2 hidden neurons. More complex network structures do not always facilitate the problems or small size networks can not produce good results for the problems. It is known that finding adequate network structure is another design problem as well as finding optimum weights in training process. For 3-Bit parity problem, BP(GD) could not find the optima in any of runs. BP(LM) could find the optimum with a success rate of 86.66%. The success rate of GA was 63.33% on this problem. The ABC algorithm was able to find the desired network output in each run with an average cycle of 179.06. For Encoder-Decoder problem, BP(GD) has 2% success and BP(LM) has 73.33 %. GA has 86.66 % and the ABC algorithm has 100% success rate on this problem. Consequently, ABC outperforms other algorithms on all problems considered in this work for the same evaluation number and can consistently find the optimum weight set for the networks. For all problems, the success rate of ABC algorithm is 100 %. An algorithm that only uses the gradient steepest descent will be trapped in a local optima, but any search strategy that analyzes a wider region will be able to cross the valley among the optima and achieve better results. In order to obtain good results for multimodal problems, the search strategy must combine the exploratory and exploitative components efficiently. Since the ABC algorithm combine the exploration and exploitation processes succesfully, it shows high performance on training feed-forward ANNs for classification problems considered in this work.

Table 4. Experimental Results 30 runs of ANN training process results produced by BP, GA and ABC Algorithms. MMSE:Mean of Mean Squared Errors of 30 Runs, SDMSE: Standard Deviation of Mean Squared Errors of 30 runs.

Algorithms		XOR6	XOR9	XOR13	3-Bit Parity	Enc. Dec.
BP (GD)	MMSE	0.1182	0.212	0.2468	0.2493	0.0809
	SDMSE	0.0763	0.0369	0.008	0.0025	0.0756
BP (LM)	MMSE	0.1107	0.0491	0.0078	0.0209	0.0243
	SDMSE	0.0637	0.0646	0.0223	0.043	0.0424
GA	MMSE	0.099375	0.047968	0.015200	0.028725	0.016400
	SDMSE	0.02785	0.052000	0.022850	0.032900	0.031300
ABC	MMSE	0.007051	0.006956	0.006079	0.006679	0.008191
	SDMSE	0.002305	0.002402	0.003182	0.002820	0.001864

Table 5. Experimental Results 30 runs of ANN training process results produced by BP, GA and ABC Algorithms. ME: Mean of Epoch Numbers, SDE: Standard Deviation of Epoch Numbers, MG: Mean of Generation Numbers, SDG: Standard Deviation of Generation Numbers, MC: Mean of Cycle Numbers, SDC: Standard Deviation of Cycle Numbers.

Algorithms		XOR6	XOR9	XOR13	3-Bit Parity	Enc. Dec.
BP (GD)	ME	31603	500	250	1600	2020
	SDE	2176	0	0	0	236.8019
BP (LM)	ME	67.53	13	9	21.333	83.3667
	SDE	59.3759	6.8304	3.2056	10.0046	174.1683
GA	MG	7500	77.067	38.6000	501.1333	400.1333
	SDG	0	33.394	25.0236	415.8687	340.4838
ABC	MC	2717.4	32	28.2	179.066666	185
	SDC	3.359377	0.182827	1.241569	12.792384	5.842378

Table 6. Success Rates of BP(GD), BP(LM), GA and ABC Algorithms

Algorithms	XOR6	XOR9	XOR13	3-Bit Parity	Enc. Dec.
BP (GD)	3	0	0	0	2
BP (LM)	6	66.66	96.66	86.66	73.33
GA	0	40	76.6667	63.3333	86.6667
ABC	100	100	100	100	100

5 Conclusion

In this work, Artificial Bee Colony Algorithm which is a new, simple and robust optimization algorithm has been used to train feed-forward artificial neural networks for classification purpose. The performance of the algorithm has been compared with the traditional back propagation algorithm and the genetic algorithm which is a well-known evolutionary algorithm. Results of the experiments

show that the Artificial Bee Colony algorithm can be successfully applied to train feed-forward neural networks. The application of ABC to other classification test problems such as iris, diabetes, cancer classification and the implementation of the algorithm for optimizing the network structure as well as optimizing weights remain as future works.

References

1. Rumelhart, D.E., Williams, R.J., Hinton, G.E.: Learning internal representations by error propagation. Parallel Distributed Processing: Explorations in the Microstructure of Cognition 1, 318–362 (1986)
2. Liu, Z., Liu, A., Wang, C., Niu, Z.: Evolving neural network using real coded genetic algorithm (GA) for multispectral image classification. Future Generation Computer Systems 20, 1119–1129 (2004)
3. Craven, M.P.: A Faster Learning Neural Network Classifier Using Selective Backpropagation. In: Proceedings of the Fourth IEEE International Conference on Electronics, Circuits and Systems, Cairo, Egypt, December 15-18, 1997, vol. 1, pp. 254–258 (1997)
4. Whitley, D., Starkweather, T., Bogart, C.: Genetic algorithms and neural networks: Optimizing connections and connectivity. Techn. Rep. CS 89-117, Department of Computer Science, Colorado State University (1989)
5. Rudnick, M.: A bibliography of the intersection of genetic search and artificial neural networks. Techn. Rep. No. CS/E 90- 001. Oregon Graduate Center, Beaverton, OR (1990)
6. Weiß, G.: Combining neural and evolutionary learning: Aspects and approaches. Techn. Rep. FKI-132-90. Institut fur Informatik, Technische Universit at Munchen (1990)
7. Weiß, G.: Towards the synthesis of neural and evolutionary learning. In: Omidvar, O. (ed.) Progress in neural networks, ch. 5, vol. 5, Ablex, Norwood, NJ (1993)
8. Albrecht, R.F., Reeves, C.R., Steele, N.C. (eds.): Artificial neural nets and genetic algorithms. In: Proceedings of the International Conference in Innsbruck, Austria. Springer, Heidelberg (1993)
9. Jones, A.J.: Genetic algorithms and their applications to the design of neural networks. Neural Computing & Applications 1, 32–45 (1993)
10. Ling, S.H., Lam, H.K., Leung, F.H.F., Lee, Y.S.: A Genetic Algorithm Based Variable Structure Neural Network, Industrial Electronics Society, 2003. In: IECON '03. The 29th Annual Conference of the IEEE, vol. 1, pp. 436–441 (2003)
11. Marshall, S.J., Harrison, R.F.: Optimization and Training of Feedforward Neural Networks By Genetic Algorithms, Artificial Neural Networks, 1991. In: Second International Conference on, November 18-20, 1991, pp. 39–43 (1991)
12. Verma, B., Ghosh, R.: A novel evolutionary neural learning algorithm, Evolutionary Computation, 2002. In: CEC '02. Proceedings of the 2002 Congress on, vol. 2, pp. 1884–1889 (2002)
13. Gao, Q., Lei, K.Q.Y., He, Z.: An Improved Genetic Algorithm and Its Application in Artificial Neural Network, Information, Communications and Signal Processing, 2005. In: Fifth International Conference on, December 06-09, 2005, pp. 357–360 (2005)
14. Yao, X.: Evolutionary artificial neural networks. International Journal of Neural Systems 4(3), 203–222 (1993)

15. Eberhart, R.C., Dobbins, R.W.: Designing Neural Network explanation facilities using Genetic algorithms. In: IEEE International Joint Conference on Publication, November 18-21,1991, vol. 2, pp. 1758–1763 (1991)
16. Jelodar, M.S., Fakhraie, S.M., Ahmadabadi, M.N.: A New Approach for Training of Artificial Neural Networks using Population Based Incremental Learning (PBIL). In: International Conference on Computational Intelligence, pp. 165–168 (2004)
17. Tsai, J.T., Chou, J.H., Liu, T.K.: Tuning the Structure and Parameters of a Neural Network by Using Hybrid Taguchi-Genetic Algorithm. IEEE Transactions on Neural Networks 17(1) (2006)
18. El-Gallad, A.I., El-Hawary, M., Sallam, A.A., Kalas, A.: Swarm-intelligently trained neural network for power transformer protection. In: Canadian Conference on Electrical and Computer Engineering, vol. 1, pp. 265–269 (2001)
19. Mendes, R., Cortez, P., Rocha, M., Neves, J.: Particle swarm for feedforward neural network training. In: Proceedings of the International Joint Conference on Neural Networks, vol. 2, pp. 1895–1899 (2002)
20. Van der Bergh, F., Engelbrecht, A.: Cooperative learning in neural networks using particle swarm optimizers. South African Computer Journal 26, 84–90 (2000)
21. Ismail, A., Engelbrecht, A.: Global optimization algorithms for training product unit neural networks. In: International Joint Conference on Neural Networks IJCNN 2000, vol. 1, pp. 132–137. IEEE Computer Society, Los Alamitos, CA (2000)
22. Kennedy, J., Eberhart, R.: Swarm Intellegence. Morgan Kaufmann Publishers, San Francisco (2001)
23. Meissner, M., Schmuker, M., Schneider, G.: Optimized Particle Swarm Optimization (OPSO) and its application to artificial neural network training. BMC Bioinformatics 7, 125 (2006)
24. Fan, H., Lampinen, J.: A Trigonometric Mutation Operation to Differential Evolution. Journal of Global Optimization 27, 105–129 (2003)
25. Ilonen, J., Kamarainen, J.I., Lampinen, J.: Differential Evolution Training Algorithm for Feed-Forward Neural Networks
26. Pavlidis, N.G., Tasoulis, D.K., Plagianakos, V.P., Nikiforidis, G., Vrahatis, M.N.: Spiking Neural Network Training Using Evolutionary Algorithms, Neural Networks, 2005. In: IJCNN '05. Proceedings 2005 IEEE International Joint Conference, vol. 4, pp. 2190–2194 (2005)
27. Plagianakos, V.P., Vrahatis, M.N.: Training Neural Networks with Threshold Activation Functions and Constrained Integer Weights, Neural Networks, 2000. In: IJCNN 2000. Proceedings of the IEEE-INNS-ENNS International Joint Conference, vol. 5, pp. 161–166 (2000)
28. Plagianakos, V.P., Vrahatis, M.N.: Neural Network Training with Constrained Integer Weights, Evolutionary Computation, 1999. In: CEC 99. Proceedings of the 1999 Congress, vol. 3, p. 2013 (1999)
29. Yu, B., He, X.: Training Radial Basis Function Networks with Differential Evolution, Granular Computing, 2006. In: IEEE International Conference on, May 10-12, 2006, pp. 369–372 (2006)
30. Yao, X., Liu, Y.: A New Evolutionary System for Evolving Artificial Neural Networks. IEEE Transactions on Neural Networks 8(3), 694–713 (1997)
31. Gao, W.: New Evolutionary Neural Networks, 2005. In: First International Conference on Neural Interface and Control Proceedings, May 26-28, 2005, Wuhan, China (2005)

32. Davoian, K., Lippe, W.: A New Self-Adaptive EP Approach for ANN Weights Training, Enformatika Transactions on Engineering, Computing and Technology, Barcelona, Spain, October 22-24, 2006, vol. 15, pp. 109–114 (2006)
33. Leung, C., Member, Chow, W.S.: A Hybrid Global Learning Algorithm Based on Global Search and Least Squares Techniques for Backpropagation Networks, Neural Networks,1997. In: International Conference on, vol. 3, pp. 1890–1895 (1997)
34. Karaboga, D.: An Idea Based On Honey Bee Swarm For Numerical Optimization, Technical Report-TR06, Erciyes University, Engineering Faculty, Computer Engineering Department (2005)
35. Basturk, B., Karaboga, D.: An Artificial Bee Colony (ABC) Algorithm for Numeric function Optimization. In: IEEE Swarm Intelligence Symposium, May 12-14, 2006, Indianapolis, Indiana, USA (2006)
36. Weiß, G.: Neural Networks and Evolutionary Computation. PartI: Hybrid Approaches in Artificial Intelligence. In: International Conference on Evolutionary Computation, pp. 268–272 (1994)
37. Fahlman, S.: An empirical study of learning speed in back-propagation networks, Technical Report CMU-CS-88-162, Carnegie Mellon University, Pittsburgh, PA 15213 (September 1988)

A Multi-supplier and Return-Policy Newsboy Model with Limited Budget and Minimum Service Level by Using GA

P.C. Yang[1], H.M. Wee[2], S.L. Chung[3], and S.H. Kang[1]

[1] Industrial Engineering and Management Department, St. John's University, Tamsui, Taipei 25135, ROC
pcyang@mail.sju.edu.tw
[2] Industrial Engineering Department, Chung Yuan Christian University, Chungli, Taiwan, 32023, ROC
[3] Information Management Department, St. John's University, Tamsui, Taipei, 25135, ROC

Abstract. A style dress outlet usually purchases products from multiple suppliers with differing cost, quality and selling price. It is assumed that some suppliers will sell their goods to the buyer outright, while some other suppliers will offer return policy for items unsold. In the latter case, the supplier buys back from the buyer the unsold items at the end of the selling season. The purpose of this study is to enable the buyer to develop a supplier selection and replenishment policy subject to limited budget. A minimal service level and uncertain market are assumed as well. Genetic algorithm (GA) is used to solve the problem.

1 Introduction

This study investigates a single order problem in which a buyer has the option of purchasing goods outright from the suppliers and/or obtaining the items with a return-policy agreement from some other suppliers. A return policy allows a buyer to return the unsold items for a partial refund. This will entice the buyer to order a larger quantity, resulting in an increase in the joint profit. Some items such as the style or catalogue goods are examples where return policies are used (Emmons and Gilbert [1]; Mantrala and Raman [2]). The "catalogue goods" are sold to customers through catalogue advertisement with fixed price during a particular selling season.

Pasternack [3] modeled a return policy and derived a global optimization in a single period with uncertain demand. He demonstrated that a return policy where a vendor offers the buyers partial credits for all unsold items could achieve channel coordination. Padmanabhan and Png [4] illustrated that the implementation of return policy can increase a vendor's profit and increase the buyer competition. Emmons and Gilbert [1] studied the effect of return policy on both the manufacturer and the buyer. Such policy is to maximize the vendor's profit by inducing the buyer to place larger order when demand is uncertain.

The importance of the single period problem increases due to the shortening products life cycle in recent years. Many extensions of the single period problem have been studied by Khouja [5]. Two major extensions are the unconstrained, single-item single-period problem, and the constrained, multi-item single-period problem. Hadley and Whitin [6] derived a constrained multi-item problem in a single period. Jucker and

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 330–340, 2007.

Rosenblatt [7] considered an unconstrained model with three types of quantity discounts: all-units quantity discount, incremental quantity discounts and Carload-lot discounts. Gerchak and Parlar [8] developed an unconstrained model in which the buyer decides on the price and the order policy. Lau and Lau [9] modeled a newsboy problem with price-dependent distribution demand. Khouja [10] developed a newsboy model in which multiple discounts are used to sell excess inventory. Khouja and Mehrez [11] extended Khouja's model [10] to consider multi-items. Lau and Lau [12] derived a capacitated multiple-product single period inventory model. Pasternack [13] developed a capacitated single-item newsboy model with revenue sharing.

GA (genetic algorithms) is a powerful tool to solve complex-structure problems with many variables. John Holland and his team applied their understanding of the adaptive processes of natural systems to design software for creating artificial systems that retained the robustness of natural systems (Holland [14]). During the last decade, GA, which is a search technique based on the mechanics of natural selection and natural genetics, has been commonly used to solve global optimization problems. Khouja, Michalewicz, and Wilmot [15], and Jinxing and Jiefang [16] studied the application of GA for solving lot-sizing problems. Mori and Tsen [17], and Li et al. [18] demonstrated that GA is effective for dealing with production planning and scheduling problems. Poulos et al. [19] derived a Pareto-optimal genetic algorithm for warehouse optimization. Zhou et al. [20] made use of GA as a tool to solve a warehouses and retailers network design. Aytug et al. [21] made a review of using genetic algorithms to solve production and operations management problems. Altiparmak et al. [22] designed a supply chain network to optimize joint total cost and service level using GA.

In this study, a single product replenished by multiple suppliers with differing cost, quality and selling price is considered in a single ordering period with return policy. GA is used to derive a supplier selection and replenishment policy under the buyer's limit budget and a minimum service level. A mathematical modeling of a newsboy problem with various constraints is derived in section 2. After illustration of GA solution procedure (section 3), a numerical example and sensitivity analysis with various budgets, service level and number of trials are carried out in section 4. The concluding remark is given in the last section.

2 Mathematical Modeling and Analysis

The model in this paper is developed on the basis of the following assumptions:

(a) Single buyer and multiple suppliers are considered. Some suppliers offer outright price and some suppliers offer price with return policy.
(b) A single item supplied by multiple suppliers has different levels of cost, quality and selling price.
(c) The demand is uncertain with known probability density function.
(d) An item with single order period, short selling season and long production lead-time is considered (an example of this type of product is the catalogue or style product).
(e) A buyer has the option of purchasing the item outright from some suppliers and/or obtaining the item through a return-policy agreement with some suppliers.
(f) Two sales priority rules: Rule 1: The items purchased with return policy begin

selling only after the outright purchase items are sold. Rule 2: The items with higher defective rate begin selling only after lower defective rate.

(g) All defective items will be found and penalized only after the items are sold to the end consumer.

(h) The buyer is subject to limited budget and minimal service level constraints.

The following notation is used:

Q_i Purchase order quantity from supplier i, i= 1, 2, 3, 4
$f(x)$ Probability density function with demand x
T Buyer's limited budget
$S.L.$ Service level
C_i Buyer's purchase cost from supplier i
d_i Product defective rate from supplier i
R_i Return price offered by supplier i for each unsold unit
P_i Selling price to the end consumer for the product from supplier i
g Buyer's unit punishment cost incurred from each defective product sold
S Buyer's unit shortage cost incurred from shortage
EP Buyer's expected profit

For simplicity, four suppliers (two suppliers with outright purchase and two suppliers with return policy) and single buyer are considered. These conditions can be changed in further research. A constrained newsboy model with four suppliers and single buyer is depicted in Figure 1.

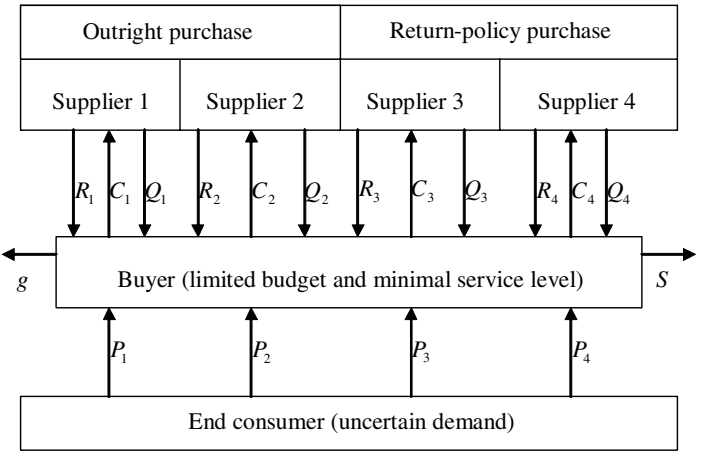


Fig. 1. A constrained newsboy model with four suppliers

Two sales priority rules are assumed. Rule 1: The items purchased with return policy begin selling only after the outright purchase items are sold. Rule 2: The items with higher defective rate begin selling only after lower defective rate. The expected sales revenue, SR can be expressed as

$$SR = \int_a^{\sum_{i=1}^{1} Q_i} P_1 x f(x)dx + \int_{\sum_{i=1}^{1} Q_i}^{\sum_{i=1}^{2} Q_i} [P_1 Q_1 + P_2 (x - Q_1)] f(x)dx$$
$$+ \int_{\sum_{i=1}^{2} Q_i}^{\sum_{i=1}^{3} Q_i} [P_1 Q_1 + P_2 Q_2 + P_3 (x - \sum_{i=1}^{2} Q_i)] f(x)dx$$
$$+ \int_{\sum_{i=1}^{3} Q_i}^{\sum_{i=1}^{4} Q_i} [P_1 Q_1 + P_2 Q_2 + P_3 Q_3 + P_4 (x - \sum_{i=1}^{3} Q_i)] f(x)dx$$
$$+ \int_{\sum_{i=1}^{4} Q_i}^{b} [P_1 Q_1 + P_2 Q_2 + P_3 Q_3 + P_4 Q_4] f(x)dx \quad (1)$$

where a and b are the lower and the upper bounds of $f(x)$.

If all the defective items are found and penalized after it is sold to the end consumer, the expected penalty cost, PC is

$$PC = \int_a^{\sum_{i=1}^{1} Q_i} d_1 g x f(x)dx + \int_{\sum_{i=1}^{1} Q_i}^{\sum_{i=1}^{2} Q_i} [d_1 g Q_1 + d_2 g (x - Q_1)] f(x)dx$$
$$+ \int_{\sum_{i=1}^{2} Q_i}^{\sum_{i=1}^{3} Q} [d_1 g Q_1 + d_2 g Q_2 + d_3 g (x - \sum_{i=1}^{2} Q_i)] f(x)dx$$
$$+ \int_{\sum_{i=1}^{3} Q_i}^{\sum_{i=1}^{4} Q_i} [d_1 g Q_1 + d_2 g Q_2 + d_3 g Q_3 + d_4 g (x - \sum_{i=1}^{3} Q_i)] f(x)dx$$
$$+ \int_{\sum_{i=1}^{4} Q_i}^{b} (d_1 g Q_1 + d_2 g Q_2 + d_3 g Q_3 + d_4 g Q_4) f(x)dx \quad (2)$$

The expected salvage value incurred from return units for unsold items at the end of selling season, SV is

$$SV = \int_a^{\sum_{i=1}^{1} Q_i} [R_1 (Q_1 - x) + \sum_{i=2}^{4} R_i Q_i] f(x)dx$$
$$+ \int_{\sum_{i=1}^{1} Q_i}^{\sum_{i=1}^{2} Q_i} \left[R_2 (\sum_{i=1}^{2} Qi - x) + \sum_{i=3}^{4} R_i Q_i\right] f(x)dx$$
$$+ \int_{\sum_{i=1}^{2} Q_i}^{\sum_{i=1}^{3} Q_i} [R_3 (\sum_{i=1}^{3} Q_i - x) + R_4 Q_4] f(x)dx$$
$$+ \int_{\sum_{i=1}^{3} Q_i}^{\sum_{i=1}^{4} Q_i} \left[R_4 (\sum_{i=1}^{4} Q_i - x)\right] f(x)dx \quad (3)$$

Shortage occurs when demand is larger than summation of Q_i. The expected shortage cost, SC is

$$SC = \int_{\sum_{i=1}^{4} Q_i}^{b} S(x - \sum_{i=1}^{4} Q_i) f(x)dx \quad (4)$$

From (1) through (4), the expected profit, EP, is the sales revenue minus penalty cost, plus salvage value, minus shortage cost and purchase cost as follows:

$$EP = SR - PC + SV - SC - \sum_{i=1}^{4} C_i Q_i \quad (5)$$

The last term in (5) is the purchase cost. The problem is a constrained nonlinear programming subject to limited budget and minimum service level, that is

$$\text{Maximum } EP = EP(Q_i) \tag{6}$$

Subject to

$$\int_0^{\sum_{i=1}^{4} Q_i} f(x)dx \geq Minimum\, S.L. \tag{7}$$

$$\sum_{i=1}^{4} C_i Q_i \leq T \tag{8}$$

and

$$Q_i \geq 0,\ i = 1, 2, 3, 4. \tag{9}$$

There are four decision variables subject to six constraints in (6) through (9).

3 GA Solution Procedure

Using a direct analogy to this natural evolution, GA presumes a potential solution in the form of an individual that can be represented by strings of genes. Throughout the genetic evolution, some fitter chromosomes tend to yield good quality offspring inherit from their parents via reproduction.

This study derives the number of deliveries per period to minimize the total cost. The objective function is $EP(Q_i)$ with decision variables Q_i. GA deals with a chromosome of the problem instead of decision variables. The values of Q_i can be determined by the following GA procedure:

(a) Representation: Chromosome encoding is the first problem that must be considered in applying GA to solve an optimization problem. Phenotype chromosome could represent a real numbers and an integer numbers here. For each chromosome, real numbers or integer numbers representation are used as follows:

$$x = Q_i, i = 1, 2, 3, 4;\ 0 \leq Q_i \leq 1000 \tag{10}$$

(b) Initialization: Generate a random population of n chromosomes (which are suitable solutions for the problem), where n=80.

(c) Evaluation: Assess the fitness $f(x)$ of each chromosome x in the population. The fitness value $f_k = f(x_k) = EP(x_k)$ where $k = 1, 2 \ldots n$

(d) Selection schemes: Select two parent chromosomes from a population based on their fitness using a roulette wheel selection technique, thus ensuring high quality have a higher chance of becoming parents than low quality individuals.

(e) Crossover: Approximately 70% crossover probability exists, indicating the probability that the parents will cross over to form new offspring. If no crossover occurs, the offspring are an exact copy of the parents.

(f) Mutation: About 30% of population mutation rate mutate new offspring at each locus (position in the chromosome). Accordingly, the offspring might have genetic material information not inherited from either parent, thus avoiding falling into the local optimum.
(g) Replacement: An elitist strategy and a steady-state evolution are used to generate a new population, which can be used for an additional algorithm run.
(h) Termination: If the number of trials exceeds 1,000,000 (or 5,000,000), then stop; otherwise go to (b).

Table 1. The evolutionary results when $T \leq \$2{,}000$ and $S.L. \geq 0.9$

Various purchases		Outright purchase		Return-policy purchase		Budget T needed Service Level Expected profit
i		i=1	i=2	i=3	i=4	
Supplier i		Supplier 1	Supplier 2	Supplier 3	Supplier 4	
Case 1	Known P_i (\$)	20	12	9	7	T=\$1,999 $S.L.$=0.900 EP=\$688.3
	Known C_i (\$)	10	9	8	7	
	Known R_i (\$)	0	0	3	4	
	Known d_i	0.02	0.05	0.1	0.2	
	Variable Q_i	118	0	0	117	
Case 2	Known P_i (\$)	20	28	26	24	T=\$1,995 $S.L.$=0.900 EP=\$2,998.0
	Known C_i (\$)	10	9	8	7	
	Known R_i (\$)	0	0	3	4	
	Known d_i	0.02	0.05	0.1	0.2	
	Variable Q_i	0	150	50	35	
Case 3	Known P_i (\$)	20	20.1	20.4	20.8	T=\$1,876 $S.L.$=0.907 EP=\$1,535.9
	Known C_i (\$)	10	9	8	7	
	Known R_i (\$)	0	0	3	4	
	Known d_i	0.02	0.05	0.1	0.2	
	Variable Q_i	1	86	49	100	
Case 4	Known P_i (\$)	20	26	28	30	T=\$1,673 $S.L.$=0.927 EP=\$3,196.1
	Known C_i (\$)	10	9	8	7	
	Known R_i (\$)	0	0	3	4	
	Known d_i	0.02	0.05	0.1	0.2	
	Variable Q_i	0	0	0	239	

Note:
Q_i is a nonnegative integer.

4 Numerical Example

The newsboy model for a buyer with uncertain demand and multiple suppliers is depicted in Figure 1, the related data are assumed as follows: limited budget $T \leq \$2{,}000$, minimal service level $S.L. \geq 0.9$, demand with uniform probability density function $f(x) = U(100,250)$, shortage cost for each shortage unit $S = \$20$, penalty cost for each defective unit $g = \$20$ and the other known parameters, P_i, C_i, R_i and d_i, are listed in Table 1. Four cases are designed for various combinations of solutions Q_i. The stopping rule is set at 5,000,000 trials for case 1 and 3, and 1,000,000 trials for case 2 and 4 depending on the status of convergence.

Using genetic algorithm, the evolutionary results of decision variables Q_i for Case 1-4 are also shown in Table 1. In Case 1, the solution is {Q_1=118 units, Q_2=0 unit, Q_3=0 unit and Q_4=117 units} because price P_1 is much greater than price P_2, cost C_4 is less than C_3, return price R_4 is greater than R_3, and the minimal service level must be met. Both the budget and service level constraints are active. In Case 2, the solution is {Q_1=0 unit, Q_2=150 units, Q_3=50 units and Q_4=35 units} because price P_2 is much greater than P_1, price P_3 greater than P_4. Both the budget and service-level constraints are active. In this case, there is budget left ($2,000-$1,995=$5) because the budget left can not afford to order any more integral unit of Q_i. In Case 3, the solution is {Q_1=1 unit, Q_2=86 units, Q_3=49 units and Q_4=100 units}. All Q_1 through Q_4 are ordered since the price difference between cases is not obvious. Both the budget and service level constraints

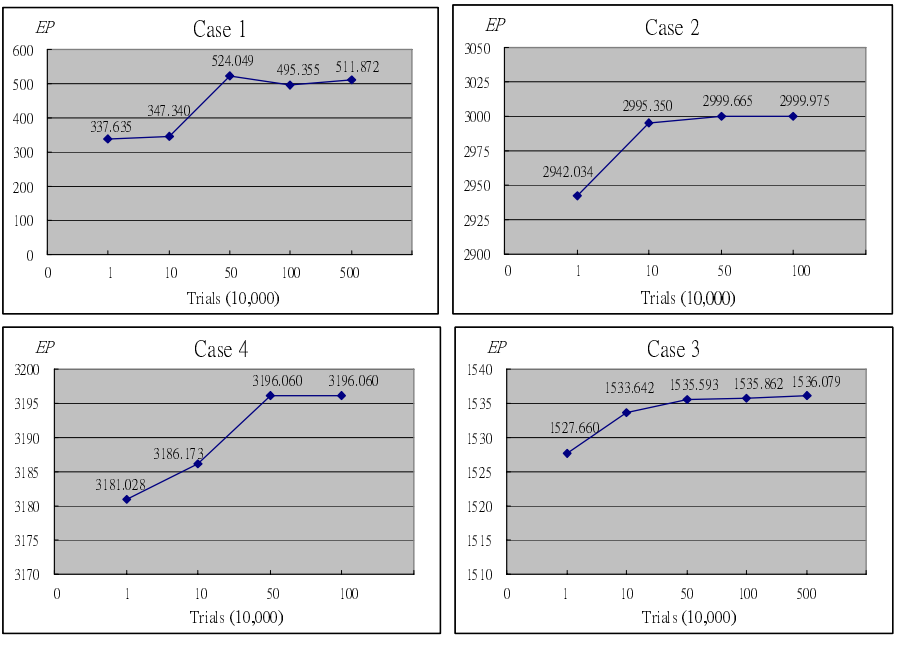


Fig. 2. Relationship between the expected profit and trials

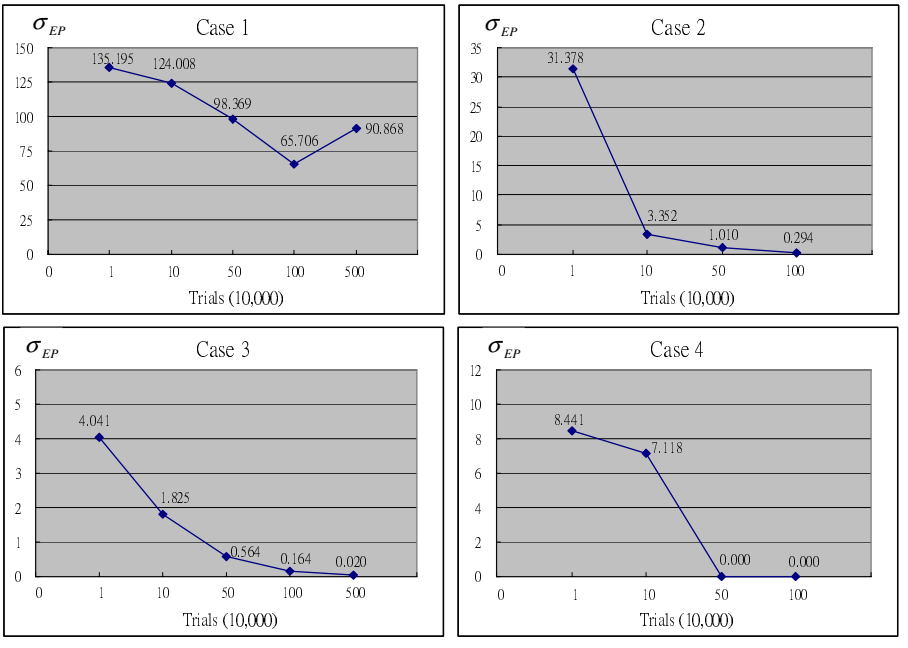


Fig. 3. Relationship between *EP*'s standard deviation and trials

Table 2. The evolutionary results when available budget is changed

Various purchases		Outright		Return-policy		*T*	*S.L.*	*EP*
Q_i		Q_1	Q_2	Q_3	Q_4			
$T \leq \$2{,}200$ and $S.L. \geq 0.9$	Case1	178	0	12	45	\$2,191	0.900	\$1,111.2
	Case2	0	150	37	51	\$2,003	0.920	\$3,000.2
	Case3	4	80	53	99	\$1,877	0.907	\$1,535.2
	Case4	0	0	0	239	\$1,673	0.927	\$3,196.1
$T \leq \$2{,}000$ and $S.L. \geq 0.9$	Case1	118	0	0	7	\$1,999	0.900	\$688.3
	Case2	0	150	50	35	\$1,995	0.900	\$2,998.0
	Case3	1	86	49	100	\$1,876	0.907	\$1,535.0
	Case4	0	0	0	239	\$1,673	0.927	\$3,196.1
$T \leq \$1{,}800$ and $S.L. \geq 0.9$	Case1	52	0	0	183	\$1,800	0.900	\$-352.0
	Case2	0	45	49	143	\$1,798	0.913	\$2,626.3
	Case3	4	66	4	162	\$1,800	0.907	\$1,499.7
	Case4	0	0	0	239	\$1,673	0.927	\$3,196.1

are non-active. In Case 4, the solution is {Q_1=0 unit, Q_2=0 unit, Q_3=0 unit and Q_4=239 units}. Only Q_4 is ordered mainly due to $P_4 < P_3 < P_2 < P_1$. Both the budget and service-level constraints are non-active.

Let the evolutionary number of trials be set at 10,000, 100,000, 500,000, 1,000,000 and 5,000,000. Run five times for each trial number, one can derive the best expected profit and the standard deviation of expected profit. The relationship between the best expected profit and the trial numbers is depicted in Figure 2 for the four cases. The relationship between the standard deviation of expected profit and the trial numbers is shown in Figure 3 for the four cases. It shows that the expected profit increases with the number of trials, and the standard deviation of expected profit decreases with the number of trials.

The sensitivity analysis is carried out when the available budget or the required minimum service level is changed in the following scenarios: {T≤\$2,200 and $S.L.$≥0.9}, {T≤\$1,800 and $S.L.$≥0.9}, {T≤\$2,000 and $S.L.$≥0.92} and {T≤\$2,000 and $S.L.$≥0.88}. The evolutionary results are shown in Table 2-3.

Table 3. The evolutionary results when the minimum service level is changed

Various purchases		Outright		Return-policy		T	$S.L.$	EP
Q_i		Q_1	Q_2	Q_3	Q_4			
$S.L.$≥0.92 and T≤2000	Case1	111	0	0	127	\$1,999	0.920	\$607.2
	Case2	0	147	40	51	\$2,000	0.920	\$3,000.1
	Case3	5	83	48	102	\$1,895	0.920	\$1,534.9
	Case4	0	0	0	239	\$1,673	0.927	\$3,196.1
$S.L.$≥0.90 and T≤\$2,000	Case1	118	0	0	117	\$1,999	0.900	\$688.3
	Case2	0	150	50	35	\$1,995	0.900	\$2,998.0
	Case3	1	86	49	100	\$1,876	0.907	\$1,535.9
	Case4	0	0	0	239	\$1,673	0.927	\$3,196.1
$S.L.$≥0.88 and T≤\$2,000	Case1	125	0	0	106	\$1,999	0.880	\$768.1
	Case2	0	149	39	49	\$1,996	0.913	\$3,000.3
	Case3	0	87	46	103	\$1,872	0.907	\$1,536.1
	Case4	0	0	0	239	\$1,673	0.927	\$3,196.1

5 Concluding Remarks

This study develops a supplier selection and the buyer's replenishment strategy for a newsboy model with limited budget and minimal service level using genetic algorithm. The buyer's optimal strategy will change based on various parameter values, the available budget and the minimum service level requirement. Sometimes mixture strategies where items are obtained by outright purchase and with return policy are

used. In other times, only a purchase with outright or return-policy is considered. The outright purchase tends to increase when the available budget increases or the required service level decreases. The return-policy purchase tends to increase when the available budget decreases or the required service level increases.

References

1. Emmons, H., Gilbert, S.M.: The role of returns policies in pricing and inventory decisions for catalogue goods. Management Science 44(2), 277–283 (1998)
2. Mantrala, M.K., Raman, K.: Demand uncertainty and supplier's returns policies for a multi-store style-good retailer. European Journal of Operational Research 115, 270–284 (1999)
3. Pasternack, B.A.: Optimal pricing and return policies for perishable commodities. Marketing Science 4(2), 166–176 (1985)
4. Padmanabhan, v., Png, I.P.L.: Returns policies: make money by making good. Sloan Management Review 37(1), 65–72 (1995)
5. Khouja, M.: The single-period (news-buyer) problem: literature review and suggestions for future research. The International Journal of Management Science 27, 537–553 (1999)
6. Hadley, G., Whitin, T.M.: Analysis of inventory systems. Prentice-Hall, Englewood Cliffs, NJ (1963)
7. Jucker, J.V., Rosenblatt, M.J.: Single-period inventory models with demand uncertainty and quantity discounts: behavioral implications and a new solution procedure. Naval Research Logistics 32, 537–550 (1985)
8. Gerchak, Y., Parlar, M.: A single period inventory problem with partially controlled demand. Computers and Operations Research 14(1), 1–9 (1987)
9. Lau, A.H.-L., Lau, H.-S.: The newsboy problem with price dependent distributions. IIE Transactions 20(2), 168–175 (1988)
10. Khouja, M.: The newsboy problem under progressive multiple discounts. European Journal of Operational Research 84, 458–466 (1995)
11. Khouja, M., Mehrez, A.: A multi-product constrained newsboy problem with progressive multiple discounts. Computers & Industrial Engineering 30, 95–101 (1996)
12. Lau, H.-S., Lau, A.H.-L.: The newsstand problem: A capacitated multi-product single-period inventory problem. European Journal of Operational Research 94, 29–42 (1996)
13. Pasternack, B.A.: The capacitated newsboy problem with revenue sharing. Journal of Applied Mathematics and Decision Sciences 5(10), 21–33 (2001)
14. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
15. Khouja, M., Michalewicz, Z., Wilmot, M.: The use of genetic algorithms to solve the economic lot size scheduling problem. European Journal of Operation Research 110, 509–524 (1998)
16. Jinxing, X., Jiefang, D.: Heuristic genetic algorithm for general capacitated lot-sizing problems. Computers and Mathematics with Applications 44, 263–276 (2002)
17. Mori, M., Tsent, C.C.: A genetic algorithm for multi-mode resource constrained project schedule problem. European Journal of Operation Research 100, 134–141 (1997)
18. Li, Y., Man, K.F., Tang, K.S.: Genetic algorithm to production planning and scheduling problems for manufacturing systems. Production Planning & Control 11(5), 443–458 (2000)

19. Poulos, P.N., Rigartos, G.G., Tzafestas, S.G., Koukos, A.K.: A Pareto-optimal genetic algorithm for warehouse multi-objective optimization. Engineering Applications of Artificial Intelligence 14, 737–749 (2001)
20. Zhou, G., Min, H., Gen, M.: A genetic algorithm approach to the bi-criteria allocation of customers to warehouses. International Journal of Production Economics 86, 35–45 (2003)
21. Aytug, H., Khouja, M., Vergara, F.E.: Use of genetic algorithms to solve production and operations management problems: a review. International Journal of Production Research 41(17), 3955–4009 (2003)
22. Altiparmak, F., Gen, M., Lin, L., Paksoy, T.: A genetic algorithm approach for multi-objective optimization of supply chain networks. Computers & Industrial Engineering 51, 197–216 (2006)

Performance Enhancement of RBF Networks in Classification by Removing Outliers in the Training Phase

Hieu Trung Huynh, Nguyen H. Vo, Minh-Tuan T. Hoang, and Yonggwan Won*

Department of Computer Engineering, Chonnam National University
300 Yongbong-dong, Buk-gu, Gwangju 500-757, Korea
hthieu@hcmut.edu.vn, ykwon@chonnam.ac.kr

Abstract. During data collection and analysis there often exist outliers which affect final results. In this paper we address reducing effects of outliers in classification with Radial Basis Function (RBF) networks. A new approach called iterative RBF (iRBF) is proposed. In which training RBF networks is repeated if there exist outliers in the training set. Detection of outliers is performed by relying upon outputs of the RBF networks which correspond to applying the training set at the input units. Detected outliers have had to be eliminated before the training set is used in the next training time. In this approach we achieve a good performance in outlier rejection and classification with training sets existing outliers.

1 Introduction

In the machine learning, one of fundamental problems is the classification. The task of classifier is to use the feature vector provided by feature extractor to assign a pattern to a class. In supervised classification, the classifier is constructed from patterns in a training set and assign label for unseen patterns from a test set. Each pattern belongs to one class, and the number of classes is known. There are many approaches for the supervised classification problem such as underlying probability densities [1], support vector machine (SVM) [2], [3], and neural networks [4]. The main problem in neural network approach is choice of activation functions and estimation of connected weights to obtain decision regions leading to minimum error. One of networks gaining much popularity in recent times is Radial Basis Function (RBF) networks [5],[6],[7],[8]; it consists of one hidden layer with radial basis activation functions. Several learning algorithms have been proposed for training RBF network [5],[6],[7],[8]. However, finding the output-layer weights is relatively simple which can use some statistical approach such as the pseudo inverse matrix or use the delta rule [9]. Selection of a learning algorithm for a particular application depends on its accuracy and speed. Literature usually focuses on more difficult problems that are identification of

* To whom all correspondences should be addressed.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 341–350, 2007.

Gaussian centers and the proper setting of the standard deviation. There are some approaches solved these problems. However most of approach does not take into account outliers existing in data collection, especially in training set it will affect so much on the classification result. Many schemes for outlier detection have proposed by researchers [10],[11],[12],[13],[14],[15],[16]. The early outlier detection methods are based-distribution. However, in practice the distribution is not easily to known. Knorr E.M et al [11] proposed a method based on distance. Their method overcame the prior knowledge of distribution but it can not work well when the data set does not have uniform density global. Jinhui Liu and Paul Gader [16] proposed outlier rejection using multilayer perceptron (MLP) and variations of the RBF network.

In this paper, we investigate the outlier rejection to improve the classification performance based on RBF networks. The outliers in the training set are detected and rejected automatically in training process, and then the parameters of the RBF network will be updated again. The experimental results show that this method brings out a good performance in outlier rejection and classification if the data set consists of outliers.

The rest of this paper is organized as follows. Section 2 reviews RBF networks. In section 3, we propose a new approach to train networks with outliers. The experimental results and analysis are shown in section 4. Finally, we conclude in the section 5.

2 RBF Networks

A popular form of basic function networks used in classifiers is depicted in Fig. 1. The pattern to be classified is passed to a set of components (basic functions), each returns a scalar value $\varphi_i, i = 1, 2, ...m$. The outputs of the network

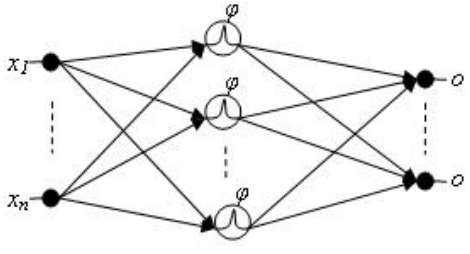


Fig. 1. The radial basic function network

are linear combination of these scalar values. Although theoretical and practical considerations indicate that the choice of the basic function is not critical, common implementations of radial basic functions which are the Gaussian bell functions described as (1).

$$\varphi_j(\boldsymbol{x}) = exp(-\frac{1}{2\boldsymbol{\sigma}_j{}^2} \| \boldsymbol{x} - \boldsymbol{\mu}_j \|^2), j = 1, \ldots, m \tag{1}$$

where m is the number of basis functions, $\boldsymbol{\mu}_j$ and $\boldsymbol{\sigma}_j$ are the j^{th} center and standard deviation respectively. For each input pattern $\boldsymbol{x}$, the neural network first present it as a vector $\boldsymbol{\varphi(x)} = [\varphi_1(\boldsymbol{x}), \varphi_2(\boldsymbol{x}), \ldots, \varphi_m(\boldsymbol{x})]$. The i^{th} output neuron is then to be computed as (2)

$$o_i = \sum_{j=0}^{m} w_{ij}\varphi_j(\boldsymbol{x}) \tag{2}$$

where w_{ij} is the weight of link from the j^{th} hidden node to the i^{th} output node, the weights w_{i0} are biases corresponding to $\varphi_0(\boldsymbol{x}) = 1$. The pattern is to be classified as belonging to the i^{th} class if its target approximates to the maximum of the outputs, $t_i \approx max_k(o_k)$.

Training of the RBF network consists of two stages, in the first stage the expectation-maximization (EM) algorithm is used to estimate the centers and deviations of components (basis functions), in the second stage the weights are estimated using back propagation style training algorithm.

3 Iterative RBF (iRBF) Approach

In the EM-stage, it is important to train components using EM updates. After completing this stage, we expect to obtain a best fit of components to be located at regions of the data set.

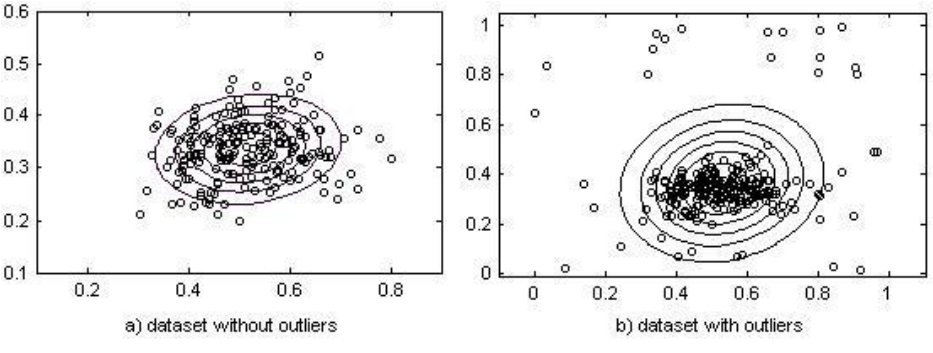


Fig. 2. Fitting of components located regions

However, it is difficult to train the components that locate well regions of the dataset if there are exist outliers in the training set. This problem can be demonstrated by two artificial data sets as shown in Fig. 2. These artificial data sets are generated from Gaussians, the number of patterns is 200 for dataset 1 (Fig. 2a) consisting of only normal patterns and 234 for dataset 2 (Fig. 2b) in which consists of 200 normal patterns from the dataset 1 and 34 outliers.

Although two data sets are the same normal patterns, trained components of them are quite different as shown in Table 1, especially their deviations. The deviation of the dataset 1 is around 10 times smaller than that of the dataset 2 which includes outliers. This difference can infer that there may have large overlapping of components among classes in the datasets containing outliers.

Table 1. Trained Components

Dataset	mean(μ)		deviaion(σ)	
Dataset 1	0.5182	0.3364	0.0095	0.0030
Dataset 2	0.5276	0.3641	0.0215	0.0264

In classification problem, a classificaion error occur when a patten $\boldsymbol{x}$ is classified as belonging to the class C_i and the true class is $C_j (i \neq j)$, the probability of error is

$$\begin{aligned} P(error) &= \sum_{j \neq i} P(\boldsymbol{x} \in C_i, C_j) \\ &= \sum_{j \neq i} P(\boldsymbol{x} \in C_i | C_j) P(C_j) \\ &= \sum_{j \neq i} \int_{C_j} p(\boldsymbol{x} | C_i) P(C_j) d\boldsymbol{x} \end{aligned}$$

This result shown that the large overlapping of components among classes gives the large probability of error. Thus, in order to increase the generalization performance of the network, it is suggested to detect and eliminate these outliers from training set and than retrain components. The outliers should not activate any output node. They are detected based on the outputs of the RBF network. If the maximal output value of the neural network is less than a rejection threshold then the corresponding input pattern is viewed as an outlier and it

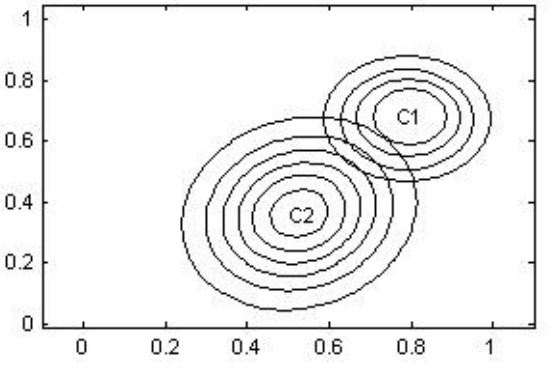


Fig. 3. The overlapping of components among classes

is rejected from the training set. Whenever parameters of components are updated, it is necessary to update weights connected from the hidden layer to the output layer. This update is easily done by using the delta rule [9] or the pseudo inverse matrix. An outline of iterative RBF training algorithm is illustrated as following:

1. Let **P** be training set
2. Train RBF network with **P**
3. Compute the output values corresponding to **P**
4. Reject patterns from **P** which have the output values less than a rejection threshold.
5. Repeat 2-4 until there are not exist outliers rejected or the number of rejected patterns larger than $pst\%$ of the data set.

The value of $pst\%$ can be set relying on the dataset so that the training set should not be less than a predefined threshold to ensure performance of training.

4 Experiments

4.1 Artificial Separable Dataset

We generate a dataset consisting of the patterns from two separable classes (class 1 and class 2) and outlier patterns. The number of patterns generated for each class is 200 for the training set and test set. Each pattern is denoted by $\boldsymbol{x} = [x_1, x_2]^T$. The normal patterns are generated from Gaussians located in [0, 1]x[0, 1], and outliers are generated from a uniform distribution located in [0, 1]x[0, 1]. An illustration of the training set is shown in Fig. 4.

The number of outlier patterns is 61 for the training set and test set. The threshold rejection is set to 0.1. If a training pattern is an outlier, both of the desired outputs are less than 0.1. In this dataset we already know which patterns

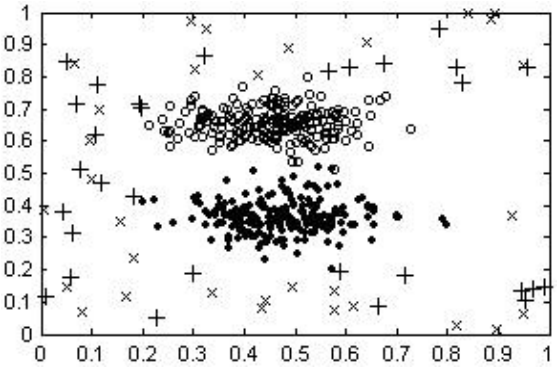


Fig. 4. The sample separable dataset

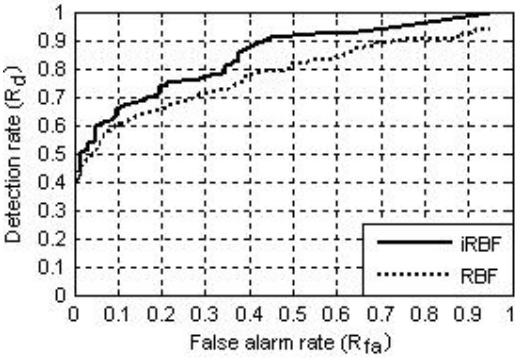


Fig. 5. The outlier rejection performance

are outliers, so we will evaluate the outlier rejection performance of our algorithm on this dataset.

As well as [16], we define a detection rate R_d as the ratio of the number of non-outlier patterns detected as non outliers to the total number of non-outliers. The false alarm rate R_{fa} is defined as the ratio of the number of outliers that are not rejected to the total number of outliers. And the classification rate R_c is defined as the ratio of the number of non-outlier patterns classified correctly to the total number of non-outlier patterns. The ROC curves used to evaluate the outlier rejection performance using RBF network and iterative RBF (iRBF) network is shown in Fig. 5. The RBF network used consists of two components (two hidden nodes).

The ROC curves indicate that, for each false alarm the detection rate using the iRBF network is better than that using RBF network or for each the detection rate the outlier rejection performance of the iRBF network is better than that of the RBF network. The classification performance of two approaches is also shown in Fig. 6. The classification performance of the iRBF network is 99.75% for all of false alarm rates.

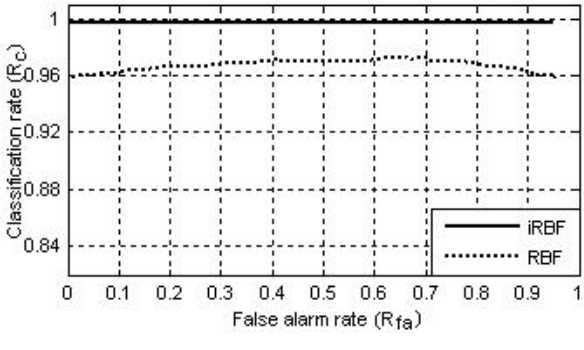


Fig. 6. The classification performance

From the ROC curves we see that the classification performance of the iRBF network is better than that of the RBF networks although the output values are the linear combination of values of the components.

4.2 Artificial Nonseparable Dataset

Similarly, we generate another dataset consisting of the patterns from two nonseparable classes (class 1 and class 2) and outlier patterns with the same parameters as in the separable case. An illustration of the training set is shown in Fig. 7. The performance of outlier rejection and classification is shown in Fig. 8 and Fig. 9. They are lower than that in the separable case. However, the performance of the iRBF network is also better than that of the RBF networks.

In this case, outliers bring about a large overlapping region which affects so much to the classification results. Finding and removing outliers give a smaller overlapping region, so it improves significantly the performance of classifications.

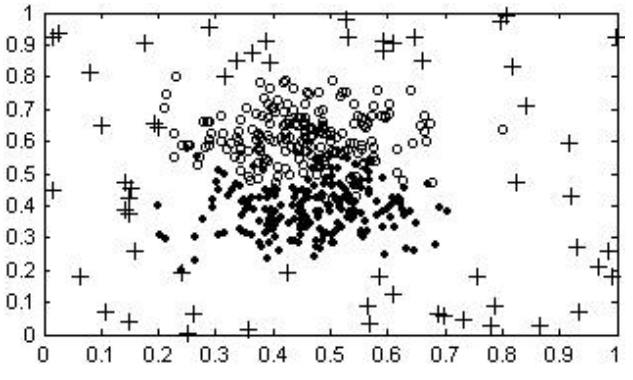


Fig. 7. The sample nonseparable dataset

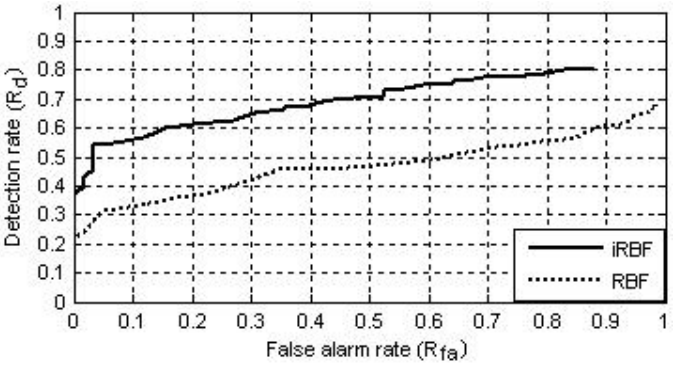


Fig. 8. The outlier rejection performance

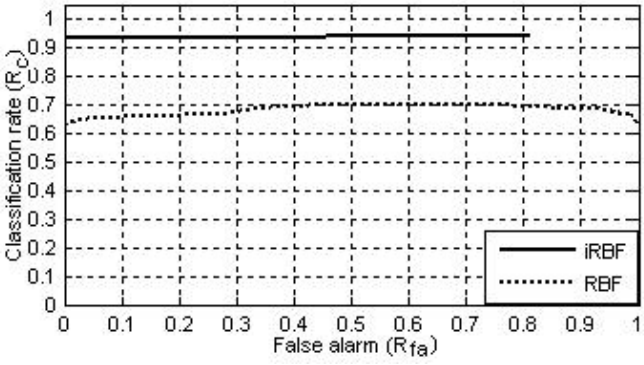


Fig. 9. The classification performance

4.3 Sonar Dataset

In this section, we evaluate the classification performance on a real data set, Sonar, Mine vs. Rocks. The dataset contains a total of 208 patterns which consists of two classes; 111 patterns for the mine class and 97 patterns for the rock class. The patterns were obtained by bouncing sonar signals off a metal cylinder at various angles and under various conditions.

Each pattern consists of 60 input features with values in the range 0.0 to 1.0 representing the energy within a particular frequency band, integrated over a certain period of time. The dataset is divided into two subsets, training set consists of 80 percent of each class, and test set consists of the remaining 20 percent.

Table 2. Classification Accuracy(%)

# of components	Training set				Testing set			
	iRBF		RBF		iRBF		RBF	
	mean	SD	mean	SD	mean	SD	mean	SD
20	82.46	1.76	82.46	1.89	80.00	5.61	78.04	4.60
30	86.89	1.37	85.02	2.56	83.14	3.50	82.68	2.92
40	88.44	2.01	88.74	2.52	84.90	3.98	83.17	2.83
50	90.96	1.63	90.96	1.63	85.37	3.67	83.63	3.71
60	93.17	2.10	92.40	1.33	85.37	3.45	85.12	3.14
70	95.21	1.01	93.95	0.91	86.34	3.67	84.15	3.50
80	95.32	1.67	95.21	1.13	86.59	3.49	84.39	4.18
90	97.31	0.90	96.82	1.05	88.54	4.30	84.39	4.90
100	97.49	1.35	97.46	1.84	88.80	3.99	86.10	3.98
110	98.56	1.02	97.60	0.86	87.56	3.89	84.88	5.24
120	99.28	0.55	99.28	0.55	88.54	2.31	87.32	2.52
130	99.64	0.56	99.32	0.63	90.24	3.89	90.00	3.98
140	99.88	0.25	99.82	0.29	89.53	3.64	89.51	3.64

In the experiment, the rejection threshold is set to 0.1. Table 2 provides experimental results averaged over 30 runs (mean and standard deviation, SD) for the iRBF and RBF networks, for several values of the number of components. These results indicate that the proposed iRBF network trained iteratively with outlier rejection provides superior performance compared to the classical RBF network.

5 Conclusions

In this paper, we propose a new approach based on the radial basis function network (RBF) to improve the performance of classification and outlier rejection. Outliers can be detected and rejected automatically in the training process. Hence, trained components of the RBF network will fit better regions of the data set. The experimental results show that our approach achieves a good performance in outlier rejection and the classification performance is significantly improved if the training set has outlier patterns that reflect their distribution.

Acknowledgment

This work was supported by grant No. RTI-04-03-03 from the Regional Technology Innovation Program of the Ministry of Commerce, Industry and Energy(MOCIE) of Korea.

References

1. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern classification. John Wiley & Sons, New York (2001)
2. Lu, S.-X., Wang, X.-Z.: A comparison among four SVM classification methods: LSVM, NLSVM, SSVM and NSVM. In: Proc. of 2004 Int'l on Machine Learning and Cybernetics, vol. 7, pp. 4277–4282 (2004)
3. Mavroforakis, M.E., Theodoridis, S.: A geometric approach to support vector machine (SVM) classification, IEEE Trans. on Neural Networks,17(3), (May 2006)
4. Lee, H.: Model selection for neural network classification. Journal of classification 18(2), 227–243 (2001)
5. Kubat, M.: Decision trees can initialize radial-basis function networks. IEEE Trans. on Neural Networks 9, 813–821 (1998)
6. Titsias, M., Likas, A.: A probabilistic RBF network for classification. In: Proc. of the IEEE-INNS-ENNS Int'l Joint Conf. on Neural Networks, vol. 4, pp. 238–243 (2000)
7. Chen, S., Cowan, C.F.N., Grant, P.M.: Orthogonal least squares learning algorithm for radial basis function networks. IEEE Trans. on Neural Networks 2, 302–309 (1991)
8. Chen, S., Chng, E.S., Alkadhimi, K.: Regularized orthogonal least squares algorithm for constructing radial basis function networks. Int'l Journal of Control 64(5), 829–837 (1996)
9. Schalkoff, R.J.: Artificial Neural Networks. McGraw-Hill, New York (1997)

10. Barnett, V., Lewis, T.: Outliers in Statistical Data, 3rd edn. John Wiley, New York (1994)
11. Knorr, E.M., Ng, R.T.: Finding Intensional Knowledge of Distance-based Outliers. In: Proc. 25th Int'l Conference on very large Data Bases, Edinburgh, Scotland, pp. 211–222 (1999)
12. Jiang, S.-Y., Li, Q.-h., Li, K.-l., Wang, H., Meng, Z.-l.: GLOF: A New Approach for Mining Local Outlier. In: Proc. of the 2nd Int'l Conference on Machine Learning and Cybernetics, pp. 157–162 (November 2003)
13. Lu, X., Li, Y., Zhang, X.: A simple strategy for detecting outlier samples in microarray data. In: The 8th Int'l Conference on Control, Automation, Robotics and Vision Kunming pp. 1331–1335 (December 2004)
14. Lu, C.-T., Chen, D., Kou, Y.: Algorithms for Spatial Outliers Detection. In: Proc. of the Third IEEE Int'l Conference on Data Mining (ICDM03) (2003)
15. Amidan, B.G., Ferryman, T.A., Cooley, S.K.: Data Outlier Detection using the Chebyshev Theorem, Aerospace, 2005. In: IEEE Conference, March 2005, pp. 1–6 (2005)
16. Liu, J., Gader, P.: Outlier Rejection with MLPs and Variants of RBF Networks. In: Proc. 15th Int'l Conference on Pattern Recognition, vol. 2, pp. 680–683 (2000)

An Evolutionary Algorithm with Diversified Crossover Operator for the Heterogeneous Probabilistic TSP

Yu-Hsin Liu, Rong-Chang Jou, Cheng-Chieh Wang, and Ching-Shu Chiu

Deptartment of Civil Engineering, National Chi Nan University
No. 1, University Road, Puli, Nantou Hsien, Taiwan 54561
{yuhsin,rcjou,s3322902,s94322509}@ncnu.edu.tw

Abstract. This paper focuses on investigating the effectiveness of the diversified crossover (DCX) operator under an evolutionary algorithm framework to solve the PTSP. Different combinations of four well-performed crossover operators for the TSP/PTSP, i.e., edge recombination (ER) crossover, order crossover (OX), order based crossover (OBX), and position based crossover (PBX), were used to investigate its effects. A set of numerical experiments were conducted to test the validity of the proposed strategy based on 90 randomly generated test instances. The numerical results showed that the DCX operator, especially by combining ER and OX crossover operators, can most effectively solve heterogeneous PTSP in most of the tested instances in comparison with the single crossover operator used in most of the previous studies. These findings show the potential of merging the proposed DCX operator into the solution framework of evolutionary algorithm, genetic algorithm or memetic algorithm for effectively solving other complicated optimization problems.

1 Introduction

The probabilistic traveling salesman problem (PTSP), a type of NP-hard problem, is a basic stochastic optimization problem [3], [15]. Due to the fact that the element of uncertainty not only exists, but also significantly affects, the system performance in many real-world transportation and logistic applications, the results from the PTSP can provide insights into research into other stochastic combinatorial optimization problems. The PTSP can be used to model many real-world applications in logistical and transportation planning, such as daily pickup-delivery services with stochastic demand, job sequencing involving changeover cost, design of retrieval sequences in a warehouse or in a cargo terminal operations, meals on wheels in senior citizen services, trip-chaining activities, vehicle routing problem with stochastic demand, and home delivery service under e-commerce [1], [2], [4], [10], [16], [26].

The PTSP is an extension of the well-known traveling salesman problem (TSP), which has been extensively studied in the field of combinatorial optimization. The goal of the TSP is to find the minimum length of a tour to all customers, given the distances between all pairs of customers. The objective of the PTSP is to minimize the expected length of the *a priori* tour where each customer requires a visit only with a given probability. The *a priori* tour can be seen as a template for the visiting sequence of all customers. In a given instance, the customers should be visited based on

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 351–360, 2007.

the sequence of the *a priori* tour while the customers who do not need to be visited will simply be skipped. The TSP can be treated as a special case of the PTSP. The main difference between PTSP and TSP is that in PTSP the probability of each node being visited is between 0.0 and 1.0 while in TSP the probability of each node being visited is 1.0.

The closed form expressions and asymptotic analysis as well as combinatorial properties for computing the a priori expected length of any given PTSP tour were first developed by Jaillet [15], [16]. Computational studies of several heuristic approaches modified from the TSP (e.g., nearest neighbor, savings approach, space-filling curve, radial sorting, 1-shift, 2-opt and 3-opt exchanges) were analyzed by Bertsimas et al. [4], Bertsimas and Howell [5], Bianchi et al. [7]. By using stochastic integer programming formulation, an exact algorithm based on an integer L-shaped method has been used to solve 50-node instances [19]. To efficiently and effectively solve the large-scale PTSP, recent studies focus on adopting new algorithmic approaches based on meta-heuristics such as ant colony optimization (ACO) [6], [9], genetic algorithm [23], simulated annealing [8], threshold accepting [26], and scatter search [20], [21], [22].

For an evolutionary computation framework which iteratively searches for the optimal solution the concept of diversification has recently been adopted to improve the effectiveness of the algorithms. For example, scatter search addressed the diversity of the reference set and used it to create the new solutions [18], [20], [21]. Variable neighborhood search (VNS) explored diversified neighborhood from the current solution [12]. Gendron et al. [11] used the diversification strategy to provide various starting solutions for the following local search phase. Ishibuchi and Shibata [14] emphasized the effects of embedding the notion of diversification on the mating scheme during the execution of evolutionary multiobjective optimization. These studies have shown the promising results of introducing diversification into the solution framework.

Inspired by the effectiveness of introducing the concept of diversification into meta-heuristics methods, this study intends to further investigate another dimension of diversification by using different combinations of crossover operators (i.e., ER, OX, OBX, PBX) under an evolutionary algorithm framework for the PTSP. To investigate the potential of incorporating the element of diversification in the crossover scheme, a set of test instances for heterogeneous PTSP are randomly generated for the purpose of comparison. The comparative results obtained can substantiate the potential of the diversified crossover (DCX) operator in solving the PTSP.

The remainder of this paper is organized as follows. Section 2 introduces the expressions for evaluating the *a priori* tour for the PTSP. Section 3 presents the details of the algorithmic procedure used in this study for the PTSP. Section 4 describes the design of the numerical experiment. Section 5 discusses the numerical results, followed by concluding comments.

2 Definition and Evaluation

The PTSP is defined on a directed graph $G := (V, E)$, where $V := \{0, v_1, v_2, ..., v_n\}$ is the set of nodes or vertices, $E \subseteq V \times V$ is the set of directed edges. Node 0 represents

the depot with the presence probability of 1.0. Each non-depot node v_i (i = 1, 2, ..., n) is associated with a presence probability p_i that represents the possibility that node v_i will be present in a given realization. Given a directed graph G, the PTSP is to find an *a priori* tour with minimal expected length in G.

Solving the PTSP mainly relies on computing the expected length of an *a priori* tour. The computation of the expected length of a specific *a priori* PTSP tour τ, denoted as $E[\tau]$, depends on the relative location of nodes on that tour and the presence probability of each node in a given instance. Jaillet and Odoni [17] proposed an efficient approach to exactly calculate $E[\tau]$ in the complexity of $O(n^3)$ for the PTSP as shown in (1).

$$E[\tau] = \sum_{i=0}^{n} \sum_{j=i+1}^{n+1} \{d_{\tau(i)\tau(j)} p_{\tau(i)} p_{\tau(j)} \prod_{k=i+1}^{j-1} (1 - p_{\tau(k)})\} \cdot \quad (1)$$

$\tau(i)$ denotes the node that has been assigned the i^{th} stop in tour τ and $p_{\tau(i)}$ is the presence probability of node $\tau(i)$. $\tau(0)$ and $\tau(n+1)$ represent node *0*, which is the depot. $d_{\tau(i)\tau(j)}$ represents the distance between nodes $\tau(i)$ and $\tau(j)$.

Even though (1) yields a polynomial evaluation time for the PTSP, the resulting $O(n^3)$ time for calculating $E[\tau]$ is still very long, especially for an evolutionary algorithm which need to repeatedly evaluate the objective function value $E[\tau]$. The proposed algorithm needs to repeatedly compare two solutions (i.e., the new solution before and after local search procedure, which is described in the next section) based on their values of $E[\tau]$. The approximate evaluation of $E[\tau]$ shown in (2) can be used to significantly increase the computation efficiency of the proposed algorithm [20].

$$E_{\lambda}^{AP}[\tau] = \sum_{i=0}^{n} \sum_{j=i+1}^{\min\{n+1, i+\lambda\}} \{d_{\tau(i)\tau(j)} p_{\tau(i)} p_{\tau(j)} \prod_{k=i+1}^{j-1} (1 - p_{\tau(k)})\} \cdot \quad (2)$$

The only difference between (1) and (2) is the choice of truncation position λ in (2). Equation (2) will have the computational complexity of $O(n\lambda^2)$, instead of $O(n^3)$ in (1). It is easy to see that (2) becomes more accurate when λ increases. A larger value of λ, however, requires more computation efforts for the computation of (2). The value of λ is set to be 10 in this study. The detailed procedure of using approximate evaluation shown in (2) to accelerate the proposed algorithm is described in the next section.

3 The Algorithmic Procedure

As shown in Fig. 1, the evolutionary algorithm proposed in this study for the PTSP mainly consists of four components. These are: initialization, selection, diversified crossover (DCX) operator, and local search.

Two loops are used in the proposed algorithm, the inner loop is denoted by I and the outer loop is denoted by G. For each generation, G, the number of iterations, I, is set to be 30 in this study. When starting to solve the PTSP, a set of initial solutions are generated based on the concept of nearest neighbor rule (NNR). The best five solutions in terms of their $E[\tau]$ values are then examined. If the best five solutions are the

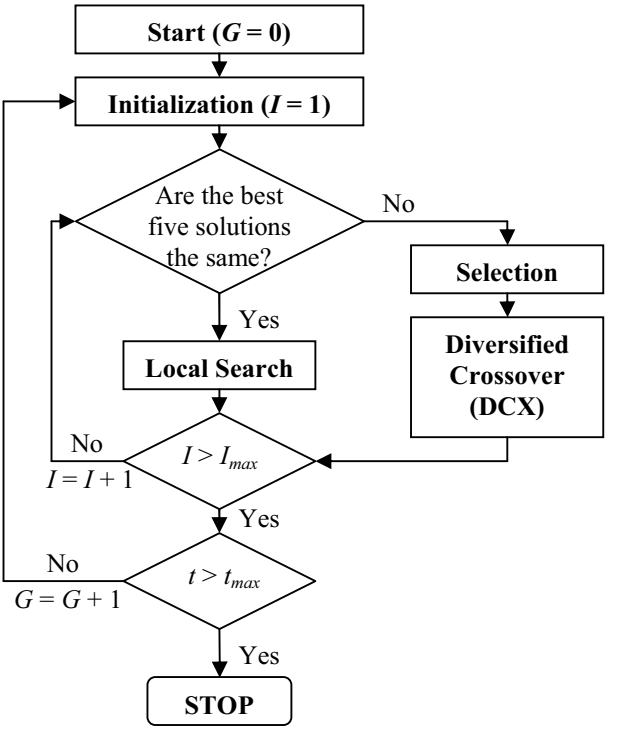


Fig. 1. The proposed solution procedure for the PTSP

same, it means that the solution is converged and the local search is used to improve the "converged" solution. Otherwise, pairs of solutions are chosen by the "selection" procedure for creating new solutions using the DCX operator. Fifteen combinations of four different crossover operators in the DCX were examined in this study. When I_{max} is reached in the current generation, the best solution of the current generation is survival to next generation to maintain the solution quality. The solutions are allowed to evolve through successive iterations and generations until the preset maximum computation time (t_{max}) is met.

3.1 Initialization

The "Initialization" is designed to generate N_{pop} initial solutions (N_{pop} = 20 in this study). Considering a PTSP with n nodes (excluding the depot, node *0*), the farthest node, a_0, from node *0* is selected first and randomly inserted into a location between ($\lfloor (n+1)/2 \rfloor$-3) and ($\lfloor (n+1)/2 \rfloor$+3). The "modified" nearest neighbor rule (NNR) [13] is used to systematically build up the sequence of the tour. Basically, the nearest, second nearest and third nearest nodes from the current node are selected by the probability of 0.8, 0.1 and 0.1, respectively. This mechanism is used to generate different initial solutions. The procedure of building the initial solution is as follows. After selecting node a_0, one of the top three nearest nodes from a_0 is randomly

selected (a_1) based on the specific probability and inserted in front of a_0. Similarly, the node (a_2) is selected and inserted behind a_0. Then, among the remaining nodes, one of the top three nearest nodes from a_1 is randomly selected (a_3) based on the specific probability and inserted in front of a_1, while node (a_4) is randomly selected based on the rule described previously and inserted behind a_2. The initial solution (tour) is thus built by following the above rule and expressed as follows.

$$0.....a_5, a_3, a_1, a_0, a_2, a_4, a_6, ...0$$

3.2 Selection

"Selection" is used to choose two tours (parents) for generating one or two new tours (children). *H1* and *H2* denote the set of candidates for parents 1 and 2, respectively, where *H1* = {the best *b* solutions} (b = 5, in this study); *H2* = {all solutions} in each iteration. Then, parents 1 and 2 are determined by randomly choosing from the sets of *H1* and *H2*, respectively. If parent 1 and parent 2 are the same, this pair will be discarded and the procedure will re-select another pair based on the same rule.

3.3 Diversified Crossover (DCX) Operator

The main purpose of the DCX is to create the new solutions using a given pair of solutions chosen by the "selection" procedure. Previous studies have investigated the performance of four commonly used crossover (i.e., ER, OX, OBX, and PBX) and found that the ER crossover is the most effective local search for the TSP and PTSP [23], [24], [25]. Inspired by the successful implementation of introducing the notion of diversification into meta-heuristics methods [11], [12], [18], instead of using the single crossover as done in most past optimization studies, the diversified crossover (DCX) operator stresses another dimension of diversification by using more than one crossover operator (i.e., ER, OX, OBX, and PBX) to create the new solutions. In the DCX operator, the best tour among parents and new solutions generated by various crossover operators (offspring) is selected to build the population of the solution for next iteration. The detailed description of ER, OX, OBX, and PBX can be found in [24].

3.4 Local Search

This component is used in an attempt to further enhance the candidate solution via a local search procedure. As stated, a couple of studies have investigated the performance of three commonly used local searches (i.e., 1-shift, 2-opt, and 3-opt) and found that the 1-shift local search is the most effective local search for the PTSP [4], [5], [7], [9]. Thus, 1-shift exchange was adopted in this study.

Assuming that one solution and its *a priori* tour is (0, v_1, v_2, v_3,..., v_n, 0) where v_i (i = 1, 2,..., n) represents the i^{th} stop of the sequence in this tour. First, a pair of nodes v_a and v_b are randomly selected from v_i (i = 1, 2,..., n). The 1-shift local search will generate a new solution by placing v_a immediately after v_b. For example, if v_a = 3 and v_b = 6 for the original tour (1, 2, 3, 4, 5, 6, 1), the new tour after 1-shift will be (1, 2, 4, 5, 6, 3, 1). Generally speaking, 1-shift local search mainly keeps the same sequence

of the original tour by only relocating one randomly selected node to the neighborhood of the other randomly selected node.

The approximate evaluation of the expected length of the *a priori* tour shown in (2) is then used to enhance the computational efficiency in the proposed evolutionary algorithm. For a specific tour τ, $E_{\lambda}^{AP}[\tau]$ is always less than the value of $E[\tau]$ because of the truncation in calculating $E_{\lambda}^{AP}[\tau]$. Let τ_b and τ_a denote the *a priori* tour before and after 1-shift local search, respectively. It means that no improvement has been found after the local search if $E_{\lambda}^{AP}[\tau] \geqq E[\tau_b]$. Equation (1) is used to exactly evaluate the solution after the local search if $E_{\lambda}^{AP}[\tau] < E[\tau_b]$. If the local search yields a better $E[\tau]$ value than the one from the original solution (i.e., $E[\tau_a] < E[\tau_b]$), the new solution (τ_a) will replace the original solution (τ_b). If no improvement has been found after the local search, no replacement will be made [20]. The above procedure is repeated N_{LS} times for each solution ($N_{LS} = 3n$ in this study).

4 Experimental Design

For heterogeneous PTSP, 90 randomly generated instances with size n = 50, 75, and 100 were used as numerical experiments in this study to examine the effects of adopting DCX strategy in evolutionary algorithm framework for the PTSP. Three groups of problem sets categorized by different intervals of customer presence probabilities were created for each problem size (n = 50, 75, and 100). Presence probabilities of customer nodes were randomly generated from a uniform distribution of intervals (0.0, 0.2), (0.0, 0.5), (0.0, 1.0), one for each problem size. The presence probability of the depot (node *0*) was assigned as 1.0. Ten different problem instances were randomly generated for each presence probability of customer nodes. For each instance, the coordinates of one depot and n customer nodes (x_i, y_i) were generated based on a uniform distribution from $[0,100]^2$. The Euclidean distance for each pair of nodes was calculated by using $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$.

A numerical experiment was designed to compare the differential effectiveness of the DCX operator in solving the PTSP. In total, fifteen tests encompassing all possible combinations of four crossover operators were conducted. Specifically, they were four tests on single crossover operators widely used in the previous studies (i.e., ER, OX, OBX and PBX) and eleven on diversified crossover operators done by using at least two crossover operators. All DCX strategies are labeled as C1-C11 accordingly, as shown in Table 1. In summary, C1-C6 represent the DCX strategy with two crossover operators; C7-C10 represent the DCX strategy with three crossover operators; C11 is the DCX strategy with all four crossover operators. In these tests, the average expected lengths of the *a priori* tour were examined. All implementations were coded in FORTRAN and performed on an Intel Pentium IV 2.8 GHz CPU personal computer with 512 MB memory, running Windows XP operating system, and using Compaq Visual FORTRAN 6.5 compiler. All fifteen tests designed in this study were used to solve each problem instance 30 times for heterogeneous PTSP in an attempt to further enhance the robustness of the results. Since ten different problem instances

were tested for each presence probability of customer nodes associated with a specific problem size, there would be a 300-run average for heterogeneous PTSP for each of the fifteen tests. The algorithm terminated when the total CPU run time exceeded the value t_{max} specified by the author according to problem size. The corresponding maximum CPU times (t_{max}) were roughly 30, 100 and 200 seconds for 50-, 75-, and 100-node instances, respectively.

Table 1. The notation of DCX strategy

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11
ER	■	■	■				■	■	■		■
OX	■			■	■		■	■		■	■
OBX		■		■		■	■		■	■	■
PBX			■		■	■		■	■	■	■

5 Computational Results and Discussion

The average optimal values of the expected length of the *a priori* tour for heterogeneous PTSP are listed in Tables 2-4. In Tables 2-4, n denotes problem size, which is the number of customer nodes; p represents the customer presence probability interval $(0.0, p]$.

For the single crossover operators used in most of the previous TSP and PTSP studies, ER and OX performed better than OBX and PBX in terms of average $E[\tau]$ value for all problem sizes and probability intervals, as shown in Table 2. Similar results can be found in TSP applications [25]. Generally, ER crossover had a better average $E[\tau]$ value than OX crossover when the presence probability intervals were large [i.e., (0.0, 0.5) and (0.0, 1.0)]. However, OX performed slightly better than ER crossover when the presence probability interval was small [i.e., (0.0, 0.2)].

For the diversified crossover (DCX) operator proposed in this study, there are six tests with 2 crossover operators (C1-C6). As shown in Table 3, the combination of ER and one other crossover operator (i.e., C1-C3) had a better average $E[\tau]$ value than the other combinations (i.e., C4-C6) when the presence probability intervals were large [i.e., (0.0, 0.5) and (0.0, 1.0)]. The combination of OBX and PBX (C6) performed the worst among these six tests (C1-C6) for all problem sizes and probability intervals. Moreover, by comparing single crossover operators and DCX with 2 crossover operators, the averaged $E[\tau]$ value yielded by the combination of ER and any of the other crossover operator (i.e., C1-C3) consistently performed better than the one yielded by using only single crossover operator. Table 4 shows the results of DCX with the combination of 3 or 4 crossover operators. The averaged $E[\tau]$ value yielded by the combination of ER and other crossover operators (i.e., C7, C8, C9, and C11) performed better than the one yielded by using only single crossover operator for almost all problem sizes and probability intervals. These results indicated that the DCX operator combined with ER and other crossover operators can yield better solution quality than by adopting a popularly used single crossover operator. However,

the combination of OX and other crossover operators performed similar results as the single crossover operator by using only OX crossover. Generally speaking, the average $E[\tau]$ value yielded by the combination of ER and OX with or without other crossover operators (i.e., C1, C7, C8, C11) performed consistently better than the one yielded by using other combinations with the DCX operator or single crossover operator.

Table 2. Results of four single crossover operators

n	*p*	ER	OX	OBX	PBX
50	0.2	230.44	**230.04**	231.39	230.69
	0.5	**340.81**	341.23	345.32	344.30
	1.0	**434.05**	436.67	443.40	443.28
75	0.2	279.55	**278.49**	280.34	279.97
	0.5	**393.88**	394.61	403.23	403.10
	1.0	**538.93**	556.45	575.31	573.37
100	0.2	311.32	**310.38**	314.97	314.23
	0.5	**457.87**	459.45	476.74	471.95
	1.0	**612.07**	630.97	650.51	648.29

Table 3. Results of DCX operator (2 crossover operators)

n	*p*	C1	C2	C3	C4	C5	C6
50	0.2	230.11	230.30	230.25	**230.08**	**230.08**	231.57
	0.5	**340.54**	341.00	340.76	341.26	341.29	345.99
	1.0	**433.01**	433.96	433.36	437.03	436.36	443.88
75	0.2	278.59	279.13	279.05	**278.33**	278.41	281.53
	0.5	392.99	393.00	**392.86**	393.82	394.45	403.57
	1.0	537.57	538.29	**537.06**	555.82	554.59	574.01
100	0.2	**309.99**	311.08	310.80	311.16	310.51	315.03
	0.5	**456.31**	457.82	457.40	461.05	461.65	474.97
	1.0	609.21	**609.05**	609.30	632.21	632.21	651.41

Table 4. Results of DCX (3 and 4 crossover operators)

n	*p*	C7	C8	C9	C10	C11
50	0.2	230.12	230.12	230.31	230.11	**230.10**
	0.5	**340.49**	340.58	340.86	341.18	340.68
	1.0	433.26	**433.24**	433.43	436.70	433.74
75	0.2	278.57	**278.48**	279.26	278.50	278.58
	0.5	**392.74**	392.79	393.09	394.64	392.84
	1.0	537.54	**536.72**	537.81	555.73	537.54
100	0.2	310.10	**310.08**	310.90	310.72	310.25
	0.5	**455.88**	456.37	457.18	461.31	457.34
	1.0	609.11	**609.04**	609.43	632.67	609.53

6 Conclusion

In this paper, the diversified crossover (DCX) operator was proposed and investigated under an evolutionary algorithm framework to solve the PTSP. Inspired by the concept of biological diversity and the successful implementation of adopting diversification into meta-heuristics methods, this study further extended to another dimension of diversification by using the combination of several crossover operators (i.e., ER, OX, OBX, and PBX crossover), instead of a single crossover operator as used in the previous TSP and PTSP studies. The numerical results show that the combination of ER and OX with or without other crossover operators yields the most promising solutions for heterogeneous PTSP. This finding further suggests the potential of merging the diversified crossover operator into the solution framework of evolutionary algorithm, genetic algorithm or memetic algorithm for effectively solving complicated optimization problems.

Acknowledgements

The paper was supported primarily as research grant funded by the National Science Council of Taiwan (NSC 95-2416-H-260-014).

References

1. Bartholdi, J.J., Platzman, L.K.: Heuristics Based on Spacefilling Curves for Combinatorial Problems in Euclidean Space. Manage. Sci. 34, 291–305 (1988)
2. Bartholdi, J.J., Platzman, L.K., Collins, R.L., Warden, W.H.: A Minimal Technology Routing System for Meals on Wheels. Interfaces 13, 1–8 (1983)
3. Bertsimas, D.: Probabilistic Combinatorial Optimization Problems. Ph.D. Thesis, Massachusetts Institute of Technology, USA (1988)
4. Bertsimas, D., Chervi, P., Peterson, M.: Computational Approaches to Stochastic Vehicle Routing Problems. Transp. Sci. 29, 342–352 (1995)
5. Bertsimas, D., Howell, L.: Further Results on the Probabilistic Traveling Salesman Problem. Eur. J. Oper. Res. 65, 68–95 (1993)
6. Bianchi, L., Gambardella, L.M., Dorigo, M.: An Ant Colony Optimization Approach to the Probabilistic Traveling Salesman Problem. In: Guervós, J.J.M., Adamidis, P., Beyer, H.-G., Fernández-Villacañas, J.-L., Schwefel, H.-P. (eds.) Parallel Problem Solving from Nature - PPSN VII. LNCS, vol. 2439, pp. 883–892. Springer, Heidelberg (2002)
7. Bianchi, L., Knowles, J., Bowler, N.: Local Search for the Probabilistic Traveling Salesman Problem: Correction to the 2-p-opt and 1-shift Algorithms. Eur. J. Oper. Res. 162, 206–219 (2005)
8. Bowler, N.E., Fink, T.M.A., Ball, R.C.: Characterization of the Probabilistic Traveling Salesman Problem. Phys. Rev. E 68, 036703 (2003)
9. Branke, J., Guntsch, M.: Solving the Probabilistic TSP with Ant Colony Optimization. J. Math. Model. Algorithm 3, 403–425 (2004)
10. Campbell, A.M.: Aggregation for the Probabilistic Traveling Salesman Problem. Comput. Oper. Res. 33, 2703–2724 (2006)

11. Gendron, B., Potvin, J.Y., Soriano, P.: Diversification Strategies in Local Search for a Nonbifurcated Network Loading Problem. Eur. J. Oper. Res. 142, 231–241 (2002)
12. Hansen, P., Mladenović, N.: Variable Neighborhood Search: Principles and Applications. Eur. J. Oper. Res. 130, 449–467 (2001)
13. Hurkens, C.A.J., Woeginger, G.J.: On the Nearest Neighbor Rule for the Traveling Salesman Problem. Oper. Res. Lett. 32, 1–4 (2004)
14. Ishibuchi, H., Shibata, Y.: Mating Scheme for Controlling the Diversity-Convergence Balance for Multiobjective Optimization. In: Deb, K., et al. (eds.) GECCO 2004. LNCS, vol. 3102, pp. 1259–1271. Springer, Heidelberg (2004)
15. Jaillet, P.: Probabilistic Traveling Salesman Problems. Ph.D. Thesis, Massachusetts Institute of Technology, USA (1985)
16. Jaillet, P.: A Priori Solution of a Traveling Salesman Problem in which a Random Subset of the Customers Are Visited. Oper. Res. 36, 929–936 (1988)
17. Jaillet, P., Odoni, A.R.: The Probabilistic Vehicle Routing Problem. In: Golden, B.L., Assad, A.A. (eds.) Vehicle Routing: Methods and Studies. North-Holland, Amsterdam, pp. 293–318 (1988)
18. Laguna, M., Martí, R.: Scatter Search: Methodology and Implementations in C. Kluwer Academic Publishers, London (2003)
19. Laporte, G., Louveaux, F., Mercure, H.: A Priori Optimization of the Probabilistic Traveling Salesman Problem. Oper. Res. 42, 543–549 (1994)
20. Liu, Y.-H.: A Scatter Search Based Approach with Approximation Evaluation for the Heterogeneous Probabilistic Traveling Salesman Problem. In: Proc. of IEEE Congr. Evolut. Comput. (CEC 2006), Vancouver, Canada (2006)
21. Liu, Y.-H.: A Hybrid Scatter Search for the Probabilistic Traveling Salesman Problem. Comput. Oper. Res. 34, 2949–2963 (2007)
22. Liu, Y.-H.: Diversified Local Search Strategy under Scatter Search Framework for the Probabilistic Traveling Salesman Problem. Eur. J. Oper. Res. (accepted after minor revisions)
23. Liu, Y.-H., Jou, R.-C., Wang, C.-C.: Genetic Algorithms for the Probabilistic Traveling Salesman Problem. In: Proc. Conf. on E-Logistics, Taoyuan, Taiwan, pp. 77–82 (2004)
24. Potvin, J.Y.: Genetic Algorithms for the Traveling Salesman Problem. Ann. Oper. Res. 63, 339–370 (1996)
25. Starkweather, T., McDaniel, S., Mathias, K., Whitley, D., Whitley, C.: A Comparison of Genetic Sequencing Operators. In: Proc. 4th Int. Conf. on Genetic Algorithms (IGGA '91). University of California at San Diego, San Diego, CA, pp. 69–76 (1991)
26. Tang, H., Miller-Hooks, E.: Approximate Procedures for the Probabilistic Traveling Salesperson Problem. Transport. Res. Rec. 1882, 27–36 (2004)

Ordered Data Set Vectorization for Linear Regression on Data Privacy

Pau Medrano-Gracia[1], Jordi Pont-Tuset[1], Jordi Nin[2], and Victor Muntés-Mulero[1]

[1] DAMA-UPC, Computer Architecture Dept.
Universitat Politècnica de Catalunya,
Campus Nord UPC, C/Jordi Girona 1-3
08034 Barcelona, (Catalonia, Spain)
{pmedrano,jpont,vmuntes}@ac.upc.edu
[2] IIIA, Artificial Intelligence Research Institute
CSIC, Spanish National Research Council
Campus UAB s/n
08193 Bellaterra (Catalonia, Spain)
jnin@iiia.csic.es

Abstract. Many situations demand from publishing data without revealing the confidential information in it. Among several data protection methods proposed in the literature, those based on linear regression are widely used for numerical data. The main objective of these methods is to minimize both the *disclosure risk* (DR) and the *information lost* (IL). However, most of these techniques try to protect the non-confidential attributes based on the values of the confidential attributes in the data set. In this situation, when these two sets of attributes are strongly correlated, the possibility of an intruder to reveal confidential data increases, making these methods unsuitable for many typical scenarios. In this paper we propose a new type of methods called *LiROP−k methods* that, based on linear regression, avoid the problems derived from the correlation between attributes in the data set. We propose the vectorization, sorting and partitioning of all values in the attributes to be protected in the data set, breaking the semantics of these attributes inside the record. We present two different protection methods: a synthetic protection method called LiROP$_s$-k and a perturbative method, called LiROP$_p$-k. We show that, when the attributes in the data set are highly correlated, our methods present lower DR than other protection methods based on linear regression.

Keywords: Privacy in statistical databases, Privacy preserving data mining, Statistical disclosure risk, Linear regression masking methods.

1 Introduction

Managing large volumes of confidential data is a common practice in any organization. In many cases, this data contains valuable statistical information required by third parties and privacy becomes essential. In this situation, it is necessary to

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 361–372, 2007.

release data preserving the statistics without revealing confidential information. This is a typical problem, for instance, in official statistics institutes.

Special efforts have been made to develop a wide range of protection methods. These methods aim at guaranteeing an acceptable level of protection of the confidential data. Specific areas such as Privacy in Statistical Databases (PSD) tackle the problem of protecting confidential data in order to publicly release it, without revealing confidential information that could be linked to an specific individual or entity. Good surveys about protection methods can be found in the literature [1,8]. Also, Privacy Preserving Data Mining (PPDM) [2] studies the case when a data mining technique allows an intruder to obtain confidential information about specific individuals.

A data set $\mathcal{D}$ can be viewed as a file with n records, where each record contains a attributes of an individual. The attributes in the original data set can be classified in two different categories, depending on their capability to identify unique individuals, as follows:

Identifiers. The identifier attributes are used to identify the individual unambiguously. A typical example of identifier is the passport number.

Quasi-identifiers. A quasi-identifier attribute is an attribute that is not able to identify a single individual when it is used alone. However, when it is combined with other quasi-identifier attributes, they can uniquely identify an individual. Among the quasi-identifier attributes, we distinguish between confidential and non-confidential, depending on whether they contain confidential information. An example of non-confidential quasi-identifier attribute would be the zip code, while a confidential quasi-identifier might be the salary.

When protecting a data set, identifiers are removed or encrypted to prevent an intruder to re-identify individuals easily. Typically, the remaining attributes are released, some of them protected. In this paper, we assume that non-confidential attributes are protected, while confidential attributes are not. This allows third parties to have precise information on confidential data without revealing who that confidential data belongs to.

Among several proposals, the use of linear regression methods is very extended. The family of synthetic data generators called *Information Preserving Statistical Obfuscation* generators (IPSO) [4] is a good example of these. The IPSO generators use multiple regression models taking confidential attributes as independent variables and non-confidential attributes as dependent variables.

In this scenario, as shown in Figure 1, an intruder might try to re-identify individuals by obtaining the non-confidential quasi-identifier data (Y) together with identifiers (Id) from other data sources. In this situation, applying record linkage between the protected attributes (Y') and the same attributes obtained from other data sources (Y), the intruder might be able to re-identify a percentage of the protected individuals together with their confidential data (X).

This attack is specially effective when the intruder uses a distance to re-identify records that takes into account correlations among attributes such as the Mahalanobis distance [11].

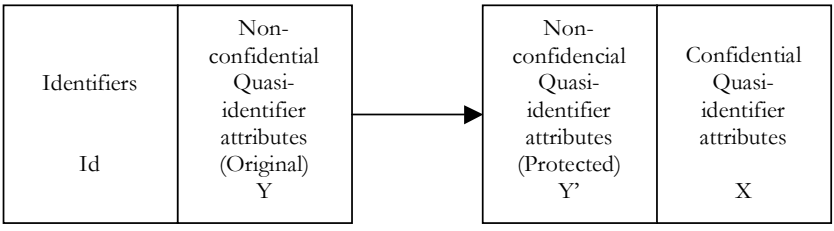


Fig. 1. Re-identification scenario

In this paper we present a new family of protection methods for numerical data called *Linear Regression on Ordered Partitions* (LiROP$-k$). The LiROP$-k$ methods ignore the possible correlation between dependent (non-confidential) (Y) and independent (confidential) (X) attributes, by eliminating the semantics of the different attributes. LiROP$-k$ methods order the whole data set in a vector, mixing attributes, in order to add perturbation by breaking the existing relations between these attributes. Then, they split the data set into several partitions of k elements, normalize the data in each partition and sort it again. After this process, an independent linear regression model is computed over each partition. All this process makes more difficult for an intruder to re-identify individuals.

Protection methods are classified depending on the philosophy used to protect data [7]. In this paper we propose two different LiROP-k methods called LiROP$_s$-k and LiROP$_p$-k, respectively. The former is a synthetic generator and the latter is a perturbative method.

We show that these methods are preferable compared to other linear regression methods when the correlation between attributes in X and Y is high. In these scenarios, the DR of LiROP-k methods is clearly lower than those obtained by IPSO methods.

This paper is organized as follows. In Section 2 we present a detailed description of the LiROP-k protection methods. In this section, we also clarify the differences between the synthetic and the perturbative variants. Section 3 presents the results. Finally, we draw some conclusions and presents some future work.

2 Linear Regression on Ordered Partitions (LiROP-k)

Linear Regression on Ordered Partitions methods (LiROP-k) propose a new vision of data handling based on the vectorization, sorting and partitioning of all the values in the data set. There are several aspects that motivate these three steps:

Vectorization. The first step is vectorization. The basic idea is to gather all the values in the data set in a single vector, independently of the attribute they

belong to. This way, we are ignoring the attribute semantics and, therefore, the possible correlation between two different attributes in the data set.

Sorting. Once all the values are inserted in the unique vector, it is necessary to sort them in order to improve the fitness of the linear regression. Note that sorting the values is a way of adding noise.

Partitioning. Even taking into account that data is sorted, calculating a unique linear regression for the whole data set is not accurate enough. In order to improve the accuracy, we propose to split the data set in several k-partitions and to calculate a linear regression for each partition. Modifying the value of k, LiROP methods allow us to tune the accuracy of the regression model by changing the size of the partition being fitted. Note that if the data set was not sorted, k would not have this property.

Since the ranges of values of two different attributes could differ significantly, the sorting step may not merge all the values coming from different attributes appropriately. For this reason, after the partitioning is complete, data is normalized in each partition and it is sorted and re-partitioned again. Data normalization improves the attribute merging and therefore, it is more difficult for an intruder to re-identify an individual.

Formally speaking, let $\mathcal{D}$ be the original data set to be protected. We denote by R the number of records in $\mathcal{D}$. Each record consists of a numerical attributes or fields. We assume that none of the registers contain blanks. We denote by N the total number of values in $\mathcal{D}$. As a consequence, $N = R \cdot a$.

Let us V be a vector of size N. LiROP$-k$ methods treat values in the data set as if they were completely orthogonal. In other words, the concepts of record and field is ignored and the N values in the data set are placed in V.

First, V is sorted increasingly. Let us denote by V_s the ordered vector of size N containing the sorted data and v_i the ith element of vector V_s, where $0 \leq i < N$.

Next, V_s is divided into smaller sub-vectors or partitions. Each sub-vector is normalized into the $[0, 1]$ interval and they are all sorted and partitioned again. We define k, where $1 < k \leq N$, as the number of values per partition. Note that if k is not a divisor of N the last partition will contain a smaller number of values. Let P be the number of k-partitions. We call r the number of values in the last partition where $0 \leq r < k$. Therefore, $N = kP + r$. If $r > 0$ the we have $P + 1$ partitions. We denote by P_m the mth partition.

Let $v_{m,n}$ be defined as the nth element of P_m:

$$\begin{cases} v_{m,n} := v_{mk+n} & n = 0 \dots k-1 \quad m = 0 \dots P-1 \\ v_{P,n} := v_{Pk+n} & n = 0 \dots r-1 \end{cases}$$

For each P_m, linear fitting is computed over the following (X, Y) points:

$$(0\,,\, v_{m,0}) \quad (1\,,\, v_{m,1}) \quad (2\,,\, v_{m,2}) \quad \cdots \quad (k-1\,,\, v_{m,(k-1)})$$

When $r > 0$, the size of the last partition (P_P) is $r < k$. In this case, the regression line of this partition is computed differently: the nearest last k points of the data set are used to compute the regression line, but only the r points

held by P_P are actually protected. This guarantees that each linear regression is computed using the same number of points, so the level of accuracy is homogeneous. Therefore, in this case, the linear fitting for the last partition is computed over the following (X, Y) points:

$$(0\,,\, v_{m,N-k}) \quad (1\,,\, v_{m,N-k+1}) \quad (2\,,\, v_{m,N-k+2}) \quad \cdots \quad (k-1\,,\, v_{m,N-1})$$

Assuming that the resulting linear regression is $l_{m,n} = \alpha_m n + \beta_m$ (where $n = 0 \ldots k-1$), then the expressions used to compute α_m and β_m are:

$$\alpha_m = \frac{2}{k(k+1)} \left[(2k-1) \sum_{n=0}^{k-1} v_{m,n} - 3 \sum_{n=1}^{k-1} n\, v_{m,n} \right]$$

$$\beta_m = \frac{2}{k(k+1)} \left[-3 \sum_{n=0}^{k-1} v_{m,n} + \frac{6}{k-1} \sum_{n=1}^{k-1} n\, v_{m,n} \right]$$

These results can be derived from the normal equations as presented in [5].

Finally, LiROP-k methods add Gaussian noise to the linear regression to partially change the order of the points. With the addition of noise, re-identification will be more difficult for an intruder even knowing the values of some attributes.

Following, we present two different methods, namely LiROP$_s$-k and LiROP$_p$-k. As we will see, the way LiROP-k adds this noise depends on the method.

2.1 LiROP$_s$-k

LiROP$_s$-k is a LiROP-k synthetic generator. As in any other synthetic generator, the values in the attributes to be protected are only used to fit the model. Once the linear regression model fits the original data, LiROP$_s$-k generates new data taking into account the model predictions and adding noise to the results.

Specifically, for each point in the regression line, a new occurrence of the Gaussian random variable (γ) is computed. The mean of γ is $\mu = 0$ and the variance $\sigma^2 = \alpha_m^2$ (i.e. $\gamma \sim N(0, \alpha_m^2)$) where, as defined previously, α_m is the slope of the regression line (for each m data partition) and the distance between two neighbor points in the X axis is 1. Consequently, the projected distance on the Y axis between two consecutive values of the regression line is $\alpha_m = \sigma$.

The protected value $p_{m,n}$ for $v_{m,n}$ is then:

$$p_{m,n} = l_{m,n} + \gamma$$

Since variance of γ is adjusted accordingly to the slope of each partition, the probability of a protected point $p_{m,i}$ to be larger than the protected point of the neighbor point $p_{m,i+1}$ when adding noise, $p_{m,i+1} - p_{m,i} < 0$, is independent of the current partition and homogeneous for all the points.

As mentioned before, this method is synthetic because the protected values are obtained from a regression model without considering the value of the particular point being masked. An example is shown in Figure 2.

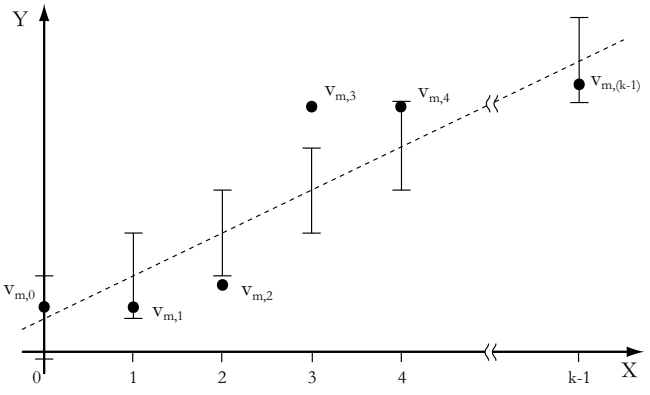


Fig. 2. LiROP$_s$-k: An example of a set of points from a partition, its linear regression and the more probable interval ($\pm\sigma$, corresponding to the centered 0.68 probability interval) for the protected value.

2.2 LiROP$_p$-k

The second LiROP-k method presented in this paper is called LiROP$_p$-k. This second method is a perturbative method and, therefore, data to be protected is not only used to fit the model, but it is also used in order to calculate the protected value generated for each point in the original data set.

In this case, the main difference between LiROP$_s$-k and LiROP$_p$-k is the method used to add noise. For LiROP$_p$-k, the mean of γ is also $\mu = 0$ but the variance is fixed at $\sigma^2 = 0.25^2$ (i.e. $\gamma \sim N(0, 0.25^2)$). The final protected value $p_{m,n}$ for $v_{m,n}$ is obtained by computing:

$$p_{m,n} = l_{m,n} + \gamma \cdot (v_{m,n} - l_{m,n})$$

This method is classified as a perturbative method because noise variance depends on the distance between the regression line and the original point.

Notice that, although the variance of γ is fixed, $\gamma \cdot (v_{m,n} - l_{m,n})$ is a Gaussian random variable with $\sigma = 0.25 \cdot (v_{m,n} - l_{m,n})$. An example is shown in Figure 3.

3 Experiments

We have presented two different protection methods based on the vectorization, sorting and partitioning of a data set.

LiROP-k methods have a very low computational cost. The execution time of any experiment presented in this section is less than 0.5 seconds using a conventional laptop. The sorting algorithm used is the quick-sort algorithm.

3.1 Data

We have considered two reference data sets for evaluation purposes proposed in the CASC project [3]. One file has been extracted using the Data Extraction

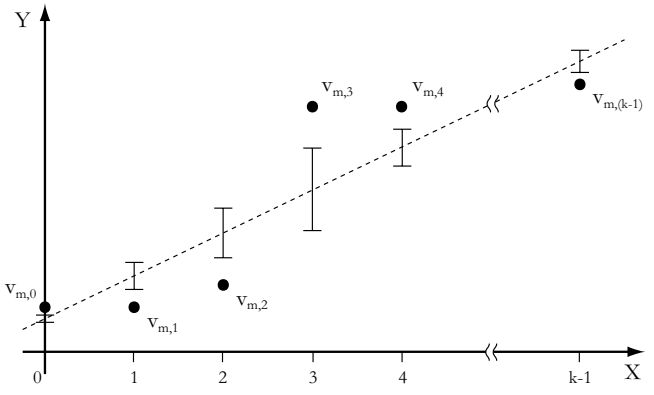


Fig. 3. LiROP$_p$-k: An example of a set of points from a partition, its linear regression and the more probable interval ($\pm 2\sigma$, corresponding to the centered 0.95 probability interval) for the protected value.

System (DES) from the U. S. Census Bureau [6], called Census and the other from the U.S. Energy Information Authority [9], called EIA.

The Census data set contains 1080 records consisting of 13 attributes (which is equal to 14040 values to be protected). Analogously to the studies presented in [8], we have selected the 13 attributes to protect the data set ignoring the relationship between the attributes. We use this data set in order to compare LiROP$_p$-k with other perturbative protection methods through an score.

The *EIA* data set, after removing the identifiers and the categorical attributes, contains 4092 records consisting of 10 attributes. These attributes are divided into two groups. The first group contains 5 non-confidential attributes which are highly correlated with the other 5 confidential attributes in the second group. The total number of values to be protected in this scenario is equal to 20460. We use this data set in order to compare LiROP-k with IPSO. In this situation, we should be able to notice a clear reduction in the DR of LiROP-k methods compared to IPSO methods.

3.2 Measures

In order to evaluate our methods, first, we compute the *score*, a typical general method used to compare two protection methods [8]. We use this score to analyze LiROP$_p$-k.

In order to calculate the *score*, we use the measures presented in previous work:

– **Information Loss (IL):** Let X and X' be matrices representing the original and the protected data set, respectively. Let V and R be the covariance matrix and the correlation matrix of X, respectively; let $\overline{X}$ be the vector of variable averages for X and let S be the diagonal of V. Define V', R',$\overline{X}'$,

and S' analogously from X'. The information loss is computed by averaging the mean variations of $X - X'$,$V - V'$,$S - S'$, and the mean absolute error of $R - R'$ and multiplying the resulting average by 100. All these measures have been extracted from [8] and are computed in the same way.

- **Disclosure Risk (DR):** We use the three different methods presented in [10] in order to evaluate DR: (i) *Distance Linkage Disclosure risk* (DLD), which is the average percentage of linked records using distance based record linkage, (ii) *Probabilistic Linkage Disclosure risk* (PLD), which is the average percentage of linked records using probabilistic based record linkage and (iii) *Interval Disclosure risk* (ID) which is the average percentage of original values falling into the intervals around their corresponding masked values. The three values are computed over the number of attributes that the intruder is assumed to know that, in our case, ranges from one to half of the attributes. These measures have been extracted from [8] and are computed in the same way.
- **Score:** A final score measure is computed by weighting the presented measures, also proposed in [8]:

$$score = 0.5\,IL + 0.125\,DLD + 0.125\,PLD + 0.25\,ID$$

Second, to compare LiROP-k and IPSO, we calculate the score and we also calculate the DR when X and Y are highly correlated using the Mahalanobis distance. The main difference between the Mahalanobis distance and the Euclidean distance is that the first takes into account the correlations among attributes. Mahalanobis distance is computed as follows:

$$d(a,b)^2 = (a-b)'[Var(V^A) + Var(V^B) - 2Cov(V^A, V^B)]^{-1}(a-b)$$

where $Var(V^A)$ is the variance matrix of the attributes of the original data, $Var(V^B)$ is the variance matrix of the attributes of the protected data and $Cov(V^A, V^B)$ is the covariance matrix between original and protected data.

Typically, the intruder does not have the information necessary in order to calculate the covariances matrix between Y and Y'. However, results in [11] show that the disclosure risk is higher when the intruder does not use the covariance matrix. In this paper, we consider both that the intruder does not know the covariance matrix (MDLD-COV0) and the intruder has hypothetically obtained this matrix somehow (MDLD).

3.3 LiROP$_p$-k Versus Other Perturbative Methods

We have tested LiROP$_p$-k with values for k ranging from 50 to 5000. The range of values for k has been selected in order to test the effect of this parameter over IL and DR. LiROP$_p$-k has been run 10 times for each value of k. The results presented in Table 1 are calculated as the average of each 10 runs obtained for each k.

Comparing results presented in [8] with our results, we observe that LiROP$_p$-k can be considered a good protection method.

Table 1. Average result of IL, DLD, PLD, ID and the score for LiROP$_p$-k using the Census data set

Census					
k	IL	DLD	PLD	ID	Score
50	0.30	47.68	88.82	99.38	42.06
200	2.13	42.95	76.81	96.96	40.28
500	3.95	40.16	61.87	91.93	37.71
700	11.67	34.27	53.11	86.40	38.36
1000	15.67	35.18	58.83	86.52	41.22
2000	16.20	36.39	59.43	87.59	41.97
3000	42.31	11.81	22.45	56.92	39.66
4000	52.43	12.84	15.17	53.51	43.09
5000	99.28	10.62	2.94	48.35	63.42

As we can observe from the table, DLD, PLD and ID measures decrease when k increases, and as we will show in the next section, this effect is maintained even when the attributes are highly correlated.

The IL of LiROP$_p$ methods increases when k increases. This makes sense because the linear regression model of each partition is forced to fit a larger number of points and, therefore, the fit error is larger, and consequently the IL increases.

3.4 LiROP-k vs IPSO for Highly Correlated Attributes

We have tested LiROP-k methods for different values of k in order to find a situation where the IL is similar to that obtained by IPSO. This way we assure a fair comparison between the two classes of methods. Specifically, k ranges from 50 to 5000. The results presented in Table 2 show the average of each 10 runs obtained for each LiROP-k method and each k, using the EIA data set. Table 3 shows the same results for the three IPSO methods.

Analyzing the figures presented in these two tables, we can observe that LiROP$_s$-k and LiROP$_p$-k present similar scores, being 28.18 and 28.29 the best scores for each method, respectively. We can also observe that these scores are lower than those obtained by IPSO methods. Specifically, IPSO-A and IPSO-B present a score of 39.05. Note that the two IPSO methods have larger values in terms of information lost (IL) and disclosure risk (DLD, PLD and ID) when compared to the best scored LiROP-k methods. The IL in IPSO-C is very high and this has a clearly negative effect on the score, that is higher than 200. From here on, we omit IPSO-C from our experiments for being its score clearly higher compared to the other two methods.

Re-identification comparison using the Mahalanobis distance. As we said in Section 1, IPSO presents a high disclosure risk when non-confidential attributes are highly correlated with the confidential attributes. In the previous

Table 2. Average results of IL, DLD, PLD, ID for LiROP-k methods using the EIA data set

	LiROP$_s$-k					LiROP$_p$-k				
k	IL	DLD	PLD	ID	Score	IL	DLD	PLD	ID	Score
50	0.82	34.32	72.40	99.80	38.70	0.82	34.35	72.40	99.80	38.70
200	3.45	19.84	28.33	97.03	32.00	3.44	19.94	28.33	96.97	32.00
500	5.05	14.97	14.65	91.79	29.18	5.12	15.10	14.65	91.85	29.24
700	12.63	12.03	4.12	79.38	28.18	12.69	12.07	4.12	79.67	28.29
1000	14.44	11.82	3.50	76.92	28.37	14.64	11.89	3.50	77.22	28.55
2000	76.44	11.50	0.68	60.16	54.78	78.74	11.47	0.68	60.42	55.99
3000	61.93	11.42	1.25	65.19	48.85	61.82	11.29	1.25	65.33	48.81
4000	54.10	11.67	0.66	63.73	44.52	55.59	11.58	0.66	63.74	45.26
5000	112.76	16.56	0.26	38.23	68.04	113.31	16.58	0.26	38.62	68.41

subsection, we compared the scores and we could see that the risk of disclosure using DLD, PLD and ID measures is higher than with our method. However, these experiments were not tested in the worst situation. We have seen that, using IPSO methods, it is possible to re-identify a large number of records using a record linkage based on the Mahalanobis distance, when the correlation between the non-confidential data (Y) and confidential attributes (X) is high, and the intruder has obtained the original values of the protected data (Y) from an alternative source.

In this subsection, we try to re-identify the protected records obtained by LiROP-k and IPSO methods using the Mahalanobis distance. We compare IPSO-A and IPSO-B with the LiROP-k methods presenting similar IL (in order to make a fair comparison), namely LiROP$_s$-1000, LiROP$_s$-2000, LiROP$_s$-3000, LiROP$_s$-4000, LiROP$_s$-5000, LiROP$_p$-1000, LiROP$_p$-2000, LiROP$_p$-3000, LiROP$_p$-4000 and LiROP$_p$-5000.

As we can observe in Table 4, the DR obtained by LiROP-k methods is significantly lower than that obtained by IPSO. While an intruder is able to re-identify 78.35 % and 78.05 % of the individuals when the data set is protected using IPSO-A and IPSO-B, respectively, it is only able to re-identify 3.16 % and

Table 3. Average results of IL, DLD, PLD, ID for IPSO methods using the EIA data set

IPSO					
Method	IL	DLD	PLD	ID	Score
IPSO-A	32.97	34.03	9.87	68.3	39.05
IPSO-B	32.98	34.01	9.90	68.29	39.05
IPSO-C	212.4	37.64	3.96	39.15	121.19

Table 4. Re-identification results using Mahalanobis distance. MDLD-COV0 stands for Mahalanobis Distance Linkage Disclosure being the covariance matrix unknown, MDLD stands for Mahalanobis Distance Linkage Disclosure being the covariance matrix known.

Method	DLD	MDLD-COV0	MDLD	PLD
IPSO-A	1.59	78.35	3.49	3.89
IPSO-B	1.59	78.05	3.30	3.89
LiROP$_s$ $k = 1000$	11.82	3.16	0.98	3.50
LiROP$_s$ $k = 2000$	11.50	0.89	0.38	0.68
LiROP$_s$ $k = 3000$	11.42	1.23	0.29	1.25
LiROP$_s$ $k = 4000$	11.67	0.71	0.17	0.66
LiROP$_s$ $k = 5000$	16.56	0.24	0.06	0.26
LiROP$_p$ $k = 1000$	11.89	3.62	1.06	3.50
LiROP$_p$ $k = 2000$	11.47	1.00	0.40	0.68
LiROP$_p$ $k = 3000$	11.29	1.35	0.79	1.25
LiROP$_p$ $k = 4000$	11.58	0.82	0.44	0.66
LiROP$_p$ $k = 5000$	16.58	0.32	0.40	0.26

3.62 % of the individuals, in the worst situation, using LiROP$_s$-k and LiROP$_p$-k, respectively. Therefore, it is preferable to use LiROP-k methods better than IPSO in this situation. Note that the results obtained by MDLD for LiROP-k methods are always similar to those obtained by PLD. This means that the intruder cannot get extra information even when data is highly correlated.

4 Conclusions and Future Work

In this paper, we have presented a new type of protection methods based on linear regressions called LiROP-k. We have shown that our methods, compared to the most well-known methods based on linear regression, are able to reduce both the score and the DR, when confidential and non-confidential attributes are correlated. We have seen that, breaking the semantics of the attributes in the records and mixing all the values in the data set by ordering them in a vector, reduces the probability of an intruder to re-identify the protected data.

As future work, we would like go further by studying data statistics and their relation with the k parameter. Our objective is to be able to estimate a right value for k instead of finding the optimal k experimentally. This would be very useful when the amount of data in the data set is very large.

Acknowledgments

The authors want to thank Generalitat de Catalunya for its support through grant number GRE-00352 and Ministerio de Educación y Ciencia of Spain for

its support through grant TIN2006-15536-C02-02. Jordi Nin wants to thank the Spanish Council for Scientific Research (CSIC) for his I3P grant.

References

1. Adam, N.R., Wortmann, J.C.: Security-Control for statistical databases: a comparative study. ACM Computing Surveys 21, 515–556 (1989)
2. Agrawal, R., Srikant, R.: Privacy Preserving Data Mining. In: Proc. of the ACM SIGMOD Conference on Management of Data, pp. 439–450 (2000)
3. Brand, R., Domingo-Ferrer, J., Mateo-Sanz, J.M.: Reference data sets to test and compare sdc methods for protection of numerical microdata. European Project IST-2000-25069 CASC (2002), http://neon.vb.cbs.nl/casc
4. Burridge, J.: Information preserving statistical obfuscation. Statistics and Computing 13, 321–327 (2003)
5. Dahlquist, G., Björck, A.: Numerical methods. Dover Publications, Mineola (2003)
6. Data Extraction System, U.S. Census Bureau: http://www.census.gov/DES/
7. Domingo-Ferrer, J., Torra, V.: Disclosure Control Methods and Information Loss for Microdata, Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, pp. 91–110. Elsevier Science, North-Holland, Amsterdam (2001)
8. Domingo-Ferrer, J., Torra, V.: A Quantitative Comparison of Disclosure Control Methods for Microdata, Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies, pp. 111–133. Elsevier Science, North-Holland, Amsterdam (2001)
9. U.S. Energy Information Authority: http://www.eia.doe.gov/cneaf/electricity/page/eia826.html
10. Torra, V., Domingo-Ferrer, J.: Record linkage methods for multidatabase data mining, Information Fusion in Data Mining, pp. 101–132. Springer, Heidelberg (2003)
11. Torra, V., Abowd, J.M., Domingo-Ferrer, J.: Using Mahalanobis Distance-Based Record Linkage for Disclosure Risk Assessment. In: Domingo-Ferrer, J., Franconi, L. (eds.) PSD 2006. LNCS, vol. 4302, pp. 233–242. Springer, Heidelberg (2006)

A Public-Key Protocol for Social Networks with Private Relationships

Josep Domingo-Ferrer

Rovira i Virgili University
UNESCO Chair in Data Privacy
Department of Computer Engineering and Mathematics
Av. Països Catalans 26, E-43007 Tarragona, Catalonia
josep.domingo@urv.cat

Abstract. The need for protecting the privacy of relationships in social networks has recently been stressed in the literature. Conventional protection mechanisms in those networks deal with the protection of resources and data, *i.e.* with deciding whether access to resources and data held by a user (owner) should be granted to a requesting user (requestor). However, the relationships between users are also sensitive and need protection: knowing who is trusted by a user and to what extent leaks a lot of confidential information about that user. The use of symmetric key cryptography to implement private relationships in social networks has recently been proposed. We show in this paper how to use public-key cryptography to reduce the overhead caused by private relationships.

Keywords: Social networks, Privacy, Private relationships, Public-key cryptography.

1 Introduction

Social networks have become an important web service [7] with a broad range of applications: collaborative work, collaborative service rating, resource sharing, searching new friends, etc. They have become an object of study both in computer and social sciences, with even dedicated journals and conferences. They can be defined as a community of web users where each network user can publish and share information and services (personal data, blogs and, in general, resources). In some social networks, users can specify how much they trust other users, by assigning them a trust level [1,6]. It is also possible to establish several types of relationships among users (for example, "colleague of", "friend of ", etc.). The trust level and the type of relationship are used to decide whether access is granted to resources and services being offered.

As pointed out in [4], the availability of information on relationships (trust level, relationship type) has increased with the advent of the Semantic Web and raises privacy concerns: knowing who is trusted by a user and to what extent discloses a lot about that user's thoughts and feelings. See [2] for an analysis of related abuses.

These privacy issues have motivated some social networks [5,8] to enforce simple protection mechanisms, according to which users can decide whether their

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 373–379, 2007.

resources and relationships should be public or restricted to themselves and those users with whom they have a direct relationship. Unfortunately, such straightforward mechanisms results in too restrictive policies.

In [3], a more flexible access control scheme is described, whereby users can be authorized to access a resource even if they have no direct relationship with the resource owner, but are within a specified depth in the relationship graph. *Access rules* are used, which specify the set of *access conditions* under which a certain resource can be accessed. Access conditions are a function of the relationship type, depth and trust level. Relationship certificates based on symmetric-key cryptography are used by a requestor to prove that he satisfies some given access conditions. To access resources held by a node with whom the requestor has no direct relationship, the requestor retrieves from a central node the chain of relationship certificates along the path from the resource owner to himself. Clearly, the central node is a trusted third party, as it knows the relationships of all nodes in the network.

In [9] a mechanism to protect personal information in social networks is described where nodes in the network are anonymous and cannot be linked to a specific users; in contrast, the data and the relationships are public, which might facilitate user re-identification.

An innovative privacy-preserving approach is described in [4] which leans on the access model in [3] and focuses on relationship protection: a user can keep private that he has a relationship of a given type and trust level with another user. Relationship certificates are encrypted and are treated like a resource in their own right: access to a certificate is granted using a *distribution rule* for that certificate, where the *distribution conditions* to be satisfied by users wishing to access the certificate are specified. If a user satisfies the distribution rule for a certificate, he receives the corresponding symmetric *certificate key* allowing him to decrypt the certificate. In [4] a scheme is proposed to manage and distribute certificate keys. Encrypted certificates are stored at a central node; due to encryption, the central node does not have access to the cleartext certificates, so it does not need to be trusted in this respect. However, the central node needs to be trusted in the following aspects: i) trust level computation when several relationship certificates are chained (indirect relationship between a resource requestor and a resource owner); ii) certificate revocation enforcement when a relationship ceases to exist (the central node must maintain a certificate revocation list and inform the other nodes about new revocations).

1.1 Contribution and Plan of This Paper

Enabling private relationships in social networks is an important issue raised in [4]. We describe in this paper a public-key protocol which offers the same features of [4] while eliminating the need for a central node.

Section 2 describes our protocol. A comparative analysis is given in Section 3. Section 4 is a conclusion.

2 A Public-Key Protocol

We follow the framework in [3], that is, we consider that the node owning a resource *rid* (hereafter, the *resource owner*) establishes an access rule $AR = (rid, AC)$ where AC is the set of access conditions to be simultaneously satisfied to access *rid*. Several alternative access rules can be defined for a resource. An access condition is a tuple $ac = (v, rt, d_{max}, t_{min})$ where v is the node with which the node requesting resource *rid* (hereafter, the *requestor*) must have a direct or indirect relationship, and rt, d_{max}, t_{min} are, respectively, the type, the maximum depth and the minimum trust level that the relationship should have. Let PK be the public key of the resource owner and SK be its private key.

We consider that access should be enforced based on the relationship path between requestor and resource owner that yields the maximum trust level. (In [3,4] the trust level is computed taking into account all paths between requestor and resource owner, which is more thorough but might lead to overprotection: a requestor with a highly trusted direct relationship to the owner might be denied access just because there is also a requestor-owner indirect relationship with low trust through a third user.)

We first describe the access control enforcement protocol.

Protocol 1 (Access control enforcement)

1. *The requestor A requests access to a resource rid owned by a resource owner B.*
2. *The resource owner B returns a signed message with all the access rules (say, $AR_1, \ldots, AR_r$) defined for rid, that is $SK(AR_1, \ldots, AR_r)$.*
3. *The requestor A sees to it that the resource owner receives one or several relationship certificates proving that the requestor satisfies all access conditions corresponding to at least one of the access rules. Several cases can be distinguished depending of the depth required:*
   (a) Depth 1. *A and B have a direct relationship, that is, A is related to B through a relationship of type rt and trust level t_{AB} represented by a tuple (A, B, rt, t_{AB}) and B is related to A through a relationship (B, A, rt, t_{BA}). Note that the relevant trust level here is t_{BA} (how much B trusts A) which is assumed to be* unknown *to A. In this case A directly asks B whether he is granted access to the resource on the basis of (B, A, rt, t_{BA}). If B evaluates that rt and t_{BA} satisfy the set of access conditions targeted by A, then A is granted access. Otherwise, A is required to resort to other direct relationships or indirect relationships.*
   (b) Depth 2. *If A and B have no direct relationships (or these are not enough to buy him access) and there are access rules with $d_{max} \geq 2$, then A asks to all users with whom A is directly related whether they have direct relationships of the relevant type rt with B. Assume C is directly related to both A and B with relationship type rt* and is willing to collaborate. *Then C sends to B a signed and encrypted certificate of his relationship $PK_B(SK_C(C, A, rt, t_{CA}))$, where $PK_B(\cdot)$ denotes encryption under the*

public key of B and $SK_C(\cdot)$ denotes encryption under the private key of C. Then C tells A that a certificate was sent to B, but does not reveal *the certificate content. At this point B evaluates whether a relationships of type rt, depth 2 and trust level $t_{CA} \cdot t_{BC}$ is enough to grant access of A to rid.*

(c) Depth 3. *If access at depth less than or equal to 2 cannot be obtained and there are access rules with $d_{max} \geq 3$, then A requests to users C directly related to him to attempt access with depth 2 on A's behalf: each C directly related to A* and willing to collaborate *contacts his other directly related users D about possible direct relationships between D and B (similarly to what A did in Step 3b). If a D with direct relationships to C and B exists and* is willing to collaborate*, B receives a chain of two certificates*

$$(PK_B(SK_C(C, A, rt, t_{CA})), PK_B(SK_D(D, C, rt, t_{DC})))$$

Now B evaluates whether a relationship of type rt, depth 3 and trust level $t_{CA} \cdot t_{DC} \cdot t_{BD}$ is enough to grant access of A to rid.

(d) Successive depths. *In case of failure at depth 3, successive depths are tried in a similar way while access rules are left which accept higher depths.*

Remark 1. There is some privacy price to be paid for using indirect relationships: intermediate users (*e.g.* C, D, etc.) are required to disclose to the resource owner B their trust level in their upstream neighbor (the upstream neighbor of C is A, the upstream neighbor of D is C, etc.). Such a disclosure brings no direct benefit to the intermediate user beyond staying in good terms with the upstream neighbor requesting collaboration. If helping the upstream neighbor is not enough motivation and/or the resource owner B is not trusted enough to be revealed how much the upstream neighbor is trusted, some intermediate nodes might refuse collaboration; this is why we stress the willingness to collaborate as a condition in Protocol 1. However, this shortcoming happens in any social network in which indirect relationships are used for resource access and users are free to decide on their collaborations. □

Remark 2. When the resource owner advertises the access rules for a resource, the access conditions in those rules leak the relationships the owner is involved in (*e.g.* if the owner accepts rt = 'Colleague at company X' this means that he works at Company X). In [4] the relationship type is kept confidential through a symmetric encryption scheme. This becomes tricky when the same relationship type is encrypted using two different keys by two different user communities and these merge at a later stage; another issue is how to revoke the key used to encrypt a given relationship type. An alternative and simpler strategy is to "camouflage" the real relationship types among a large number of bogus relationships; then access conditions are published some of which use real relationship types and most of which use bogus relationship types. A bogus relationship type rt' is one that has never been established by the owner with anyone, so that no

one can request access based on rt'. The advantage is that a snooper cannot tell bogus relationships from real ones, so that he does not know which relationships the owner is actually involved in. □

We will show in the next section that Protocol 1 has the advantage of not requiring any complementary protocols for certificate management or revocation, nor does it require to use any trusted third party or central node.

3 Comparative Analysis

We will compare Protocol 1 with the proposal in [3,4] based on two aspects: protection of relationship certificates and certificate revocation.

3.1 Protection of Relationship Certificates

In the proposal [3,4] relationships are protected through relationship certificates to be presented by a requestor to prove that he has the credentials to access a certain resource. Such certificates are stored at a central node and retrieved from it when needed.

Relationship certificates must be protected because they often convey information that the nodes involved in the relationship would like to keep private (who is related with whom, what the relationship type is and what the trust level is). The authors of [4] propose to view a certificate as a resource and define *distribution rules* and *distribution conditions* to access it; in fact distribution rules and conditions are analogous to the access rules and conditions defined for standard resources. Each certificate is kept encrypted at the central node (who cannot thus read it) under a *certificate key* $CK = (k_{RC}, id_{RC})$ where k_{RC} is a symmetric key and id_{RC} is the corresponding key identifier. When a user satisfies the distribution rule for a certificate, he is given the corresponding certificate key.

The following features of the above model leave room for improvement:

- A central node is needed which might not always be reachable, especially in social networks implemented over ad hoc networks.
- Being symmetrical, certificate keys require a complex key management scheme. In fact, a certificate key distribution algorithm is proposed in [4]: each time a node (user) establishes a new relationship, or when the node receives a new certificate key from one of its neighbors, the algorithm is run by the node to verify whether such a key must be further distributed. Distribution rules related to the certificate are transformed to be relative to the receiving node: the algorithm substitutes the node components of each distribution condition with the identifier of the node receiving the rule, and the depth components are decreased by one unit.

The improvement offered by Protocol 1 can be described as:

Fault tolerance: No central node is required to be reachable all the time. For each resource access, a user tries to get the backing of the nodes with whom

he is related. If one of those nodes is temporarily unreachable, this is not a problem as long as the user can reach other nodes. As mentioned above, access is enforced based on the (reachable) relationship path between requestor and resource owner that yields the maximum trust level; since we do not need all relationship paths between requestor and owner, a certain amount of connectivity failure is tolerable.

TTP-freeness: No central node is required to compute the trust level between requestor and owner. This computation is done by the resource owner on the basis of the received certificates (whereas in [4] the central node is entrusted with such a critical computation).

Minimum relationship disclosure: In Protocol 1 the resource requestor does not see any of the relationship certificates that will be used by the resource owner to decide whether he is granted access. The operation of [4] is different, because there it is the requestor who retrieves the relevant relationship certificates (the certificate chains connecting the requestor to the owner) and forwards them to the owner. This is undesirable: if the retrieved certificates are left unencrypted, then the requestor can see the private information in them; if they are encrypted, then a scheme for managing certificate keys is required (see remarks about certificate key distribution).

3.2 Certificate Revocation

In a social network, users must be able to establish new relationships and revoke some of the existing ones. The use of certificate keys in [4] has implications for revocation. If certificate revocation is not properly managed, unauthorized distribution of certificate keys may occur: if A revokes a certificate relationship with depth 2 relating A to B, it might happen that B continues to distribute the certificate key to his neighbors after revocation. The solution adopted in [4] is to notify revocation to the central node, who removes the revoked certificate from the central certificate directory and stores the certificate identifier in a certificate revocation list (CRL). Thus, the central node actually plays the role of a trusted third party. Besides TTPs being a potential source of conflict, this arrangement requires all users to check the CRL before accepting or distributing a certificate, which causes substantial overhead and implicitly assumes some user trustworthiness.

Protocol 1 has the advantage of being TTP-free. If A wants to access a resource owned by B using an indirect relationship through C and C has decided to revoke his relationship with A, then C will simply send no certificate to B. There is no need to keep revocation lists at a central node.

4 Conclusion and Future Work

Protecting the type and trust level of relationships is an important privacy issue in social networks first raised in the important contribution [4]. We are aware of only one approach in the literature addressing that problem. Being based

on symmetric-key cryptography, that approach requires a central node which must always be reachable and which behaves as a trusted third party. We have presented a public-key protocol that also achieves relationship protection, with the advantages of being fault-tolerant and TTP-free. Besides, the new protocol avoids revealing the content of relationships to the resource requestor and substantially simplifies relationship revocation.

Future work will focus on devising public-key cryptographic techniques to deal with the relationship type leakage associated to access rule advertising by owners. The aim is to gain simplicity with respect to the symmetric-key strategy without having to "litter" access rules with a large number of bogus relationship types.

Disclaimer and Acknowledgments

The author is solely responsible for the views expressed in this paper, which do not necessarily reflect the position of UNESCO nor commit that organization. This work was partly supported by the Spanish Ministry of Education through project SEG2004-04352-C04-01 "PROPRIETAS" and by the Government of Catalonia under grant 2005 SGR 00446.

References

1. Ashri, R., Ramchurn, S.D., Sabater, J., Luck, M., Jennings, N.R.: Trust evaluation through relationship analysis. In: 4th International Joint Conference on Autonomous Agengs and Multiagent Systems (AAMAS 2005), pp. 1005–1011. ACM, New York (2005)
2. Barnes, S.B.: A privacy paradox: social networking in the United States, First Monday, vol. 11(9) (September 2006)
3. Carminati, B., Ferrari, E., Perego, A.: Rule-based access control for social networks. In: Meersman, R., Tari, Z., Herrero, P. (eds.) On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops. LNCS, vol. 4278, pp. 1734–1744. Springer, Heidelberg (2006)
4. Carminati, B., Ferrari, E., Perego, A.: Private relationships in social networks (IEEE Catalog Number 07EX1627). In: Private Data Management'2007 workshop of the ICDE 2007- 23rd IEEE International Conference on Data Engineering, Istanbul, Turkey, April 15-20, 2007, IEEE Computer Society Press, Los Alamitos (2007)
5. Facebook: http://www.facebook.com
6. Sabater-Mir, J.: Towards the next generation of computational trust and reputation models. In: Torra, V., Narukawa, Y., Valls, A., Domingo-Ferrer, J. (eds.) MDAI 2006. LNCS (LNAI), vol. 3885, pp. 19–21. Springer, Heidelberg (2006)
7. Staab, S., Domingos, P., Mika, P., Golbeck, J., Ding, L., Finin, T.W., Joshi, A., Nowak, A., Vallacher, R.R.: Social networks applied. IEEE Intelligent Systems 20(1), 80–93 (2005)
8. Videntity: http://videntity.org
9. Wang, D.-W., Liau, C.-J., Hsu, T.S.: Privacy protection in social network data disclosure based on granular computing. In: Proc. of the 2006 IEEE International Conference on Fuzzy Systems (HICSS'05), pp. 997–1003. IEEE Computer Society, Los Alamitos (2006)

An Incentive-Based System for Information Providers over Peer-to-Peer Mobile Ad-Hoc Networks

Jordi Castellà-Roca, Vanesa Daza, Josep Domingo-Ferrer, Jesús A. Manjón, Francesc Sebé, and Alexandre Viejo

Rovira i Virgili University
UNESCO Chair in Data Privacy
Department of Computer Engineering and Mathematics
Av. Països Catalans, 26, E-43007 Tarragona, Catalonia
{jordi.castella,vanesa.daza,josep.domingo,jesus.manjon}@urv.cat
{francesc.sebe,alexandre.viejo}@urv.cat

Abstract. An architecture for a peer-to-peer mobile ad-hoc network offering distributed information provision is presented. Any user can volunteer to become an information server (a server-user). Volunteering implies devoting some of the user's computational resources (storage, bandwidth, processing power) to serving information. An incentive scheme is proposed to encourage end-users to become server-users. The latter are rewarded proportionally to the number of end-user queries served. The proposed architecture is specified as a protocol suite taking security and privacy aspects into account. Details are given on an implementation completed on a WiFi ad-hoc network for the specific case of a distributed tourist information service.

Keywords: Peer-to-peer networks, Mobile ad-hoc networks, Distributed information servers, Security and privacy.

1 Introduction

The new-generation mobile devices (*e.g.* cell phones, PDAs ...) are enabled with wireless communications technologies like IEEE 802.11b WiFi, which paves the way to a broad range of services based on ad-hoc networks, *i.e.* spontaneous networks without an underlying infrastructure. Clearly, a wireless ad-hoc network spanning a local or a metropolitan area is an extremely flexible and powerful tool, allowing extensive peer-to-peer communication. Access points in these new wireless networks can be viewed as peers whose number and geographical span can be much larger than what is affordable with conventional peer-to-peer wired networks. From the topology standpoint, the ad-hoc model is particularly well suited to set up hop-by-hop communication to end-nodes connected to such wireless peer-to-peer networks.

In this paper, we propose an information system aiming to provide information *just in time* and *just in place* in a specific area. Specifically in our scenario,

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 380–392, 2007.

the system is deployed in a city within which a user, regardless of her location, can request information any time using her WiFi mobile device and its associated software. As justified above, we assume that access to information is made possible by a metropolitan ad-hoc network based on WiFi peers.

A typical application would be tourist information: a person touring a city can query the system to obtain a list of museums near her current location or some information about a given historical building she is currently visiting.

The system we have just sketched needs an architecture with several information servers replicating the knowledge, because a single information server could become a bottleneck due to downloads of large pieces of multimedia content (pictures, videoclips, etc.) or due to temporary connectivity loss caused by dynamic changes in the topology of the ad-hoc network. However, using several servers poses the problem of the cost associated to installing a large number of servers around the city plus the costs resulting from their subsequent administration.

Instead of incurring the expense of installing and maintaining many servers, we propose to trust individual users to offer services to the WiFi ad-hoc network like in classic peer-to-peer communities (*e.g.* BitTorrent, Gnutella, etc). To keep the network running efficiently, we suggest to properly motivate the users to devote some of their computing resources to providing those services.

1.1 Previous Work

Providing and exchanging information between users is a research topic heavily addressed in the literature. At the beginning, the information-sharing systems followed a client-server paradigm (*e.g.* Napster) with the bottleneck problems associated to this kind of structure. Afterwards, real peer-to-peer networks arose, and decentralized content-sharing applications appeared, which avoided the client-server related problems. Examples of this approach are BitTorrent or Gnutella.

Peer-to-peer applications are often deployed in traditional wired networks, although most of them could also work on wireless ad-hoc networks. In [6] some applications working on *mobile* ad-hoc networks are presented. Within the scope of mobile peer-to-peer networks, we can find the iClouds Project [5] which investigates several kinds of collaboration among mobile users using the hop-by-hop communication paradigm related to ad-hoc networks.

As stated in [16], the motivation of users to participate in the community is a crucial factor for the success of a peer-to-peer system. The authors of that paper describe various methods to motivate different kinds of users and describe the design of a peer-to-peer system called Comutella, which has been developed for supporting file and service sharing. [4] presents I-Help, a system devoted to sharing help between students where participation can be rewarded either in real dollars (for paying teaching assistants) or in marks improvement (for students). This system also uses hired knowledgeable persons, teaching assistants or lecturers, to be constantly on-line and to immediately answer any question. This kind of users can be considered knowledge providers in this system, and they are very similar to the information providers presented in our proposal.

Providing motivation to nodes, who offer services to end-users, implies the need of a secure way for collecting the rewards from the served users. Secure electronic payment is a profusely studied research topic. From electronic money to e-coupons [1], there are several electronic payment methods suitable for mobile devices. However, for the specific case of secure incentive-based schemes, the literature is rather scarce. [13] and [17] propose incentive-based schemes where the network nodes have an account and the content provider gives them credit depending on the information they have uploaded. The network nodes can use their credit to increase their download rate or change it for money. Nonetheless, these proposals are not designed for a mobile ad-hoc network, and the security is only focused toward the protection of the copyrighted content. Thus, the credit of the network node can be tampered with. In [10], the authors describe and analyze three different schemes aimed at giving proofs of service to providers in a hybrid environment where a server collaborates with a peer-to-peer network intended to mitigate server congestion. The authors compare the three schemes in terms of scalability, effectiveness and cost. Recently, [11] have proposed a lightweight and hash-chain-based micropayment scheme for ad-hoc networks. They compensate collaborative peers that sacrifice their resources to relay packets for others. They also use an offline trusted third party (the Private Key Generator), but following an ID-based approach instead of our PKI-based one.

In our proposal, we provide proofs of service to information providers using a scheme similar to the scenario of fair exchange with an offline trusted third party (TTP) described in [10]. However, our scenario is pretty different since we deal with a pure peer-to-peer community working through a mobile ad-hoc network.

1.2 Contribution and Plan of This Paper

In this paper we focus on an environment where some end-user nodes build a WiFi ad-hoc network and act as information providers. These nodes devote some of their computational resources (storage, bandwidth, processing power) to storing and serving information. In this way, when another user in the ad-hoc network requests some information, those nodes storing the requested information can supply it. Volunteering to become an information provider is rewarded depending to the amount of served information requests. Information is shared according to the peer-to-peer paradigm.

Note. We concentrate on the WiFi technology because it currently is the most common choice in mobile ad-hoc networks. However, the system described here is open to similar wireless technologies.

To perform information searches through the peer-to-peer network, we have implemented a catalog-based search engine. Our system makes use of an item catalog shared between information providers. Since updating a unique catalog distributed among all peers is a resource- and time-consuming task, this approach is best suited to a system where the items change rarely. We have taken as a case study a tourist information system: the information on accommodation, monuments, etc. is pretty stable. For items that change more dinamically, *e.g.*

films shown in theaters in a specific moment, a different search engine should be used. Anyway, the communication protocol in our system has been designed to be independent of the search engine used, which can be replaced if necessary.

Every time a server provides information to some user, it obtains a receipt allowing it to prove that it has performed the service. Periodically, the server contacts the main content provider (*i.e.* the main source of content, in our case study the city tourist office) to get paid according to the number of requests it has served. This is a way to encourage users to devote more resources to providing information. The more information a server stores, the more requests it will be able to serve, and thus, the more money it will receive for its service. A server located at a certain place will probably contain information that may interest nearby users.

Section 2 describes the architecture of the proposed system. Section 3 is a security and privacy analysis. Implementation details are given in Section 4. Section 5 is a conclusion.

2 System Architecture

In this section we present the system components: entities, messages exchanged between entities and protocols between entities. We note that the system architecture is similar to a multi-agent system.

2.1 Entities

The proposed system consists of the following entities:

- **Content Source (CS):** This is the entity offering the information service. In the aforementioned example about tourist information, this entity may be the tourist office of a city holding information of particular interest for residents or visitors. Some examples could be:
  - Information on historical landmarks, including short multimedia videos, audio streams and digital documents on them.
  - Schedule of cultural and leisure activities, like cinema or theater, including trailer viewing options.
  - Information about restaurants: opening hours, menus, prices.
  - Location of services: police stations, hospitals, pharmacies.
- **Users:** Users whose devices form the ad-hoc network. We distinguish two kinds of users:
  - Those that query the system when they need information. Normally, they use a mobile device and request information through the ad-hoc network. We refer to them as *end-users (EU)*.
  - Those that devote part of their computational resources to storing and serving some of the information supplied by the content source. These users not necessarily use a mobile device. They could store information in their desktop PC with an ad-hoc network interface. We refer to these users as *server-users (SU)*.

Our description assumes an underlying information sharing system (following the P2P paradigm) like those described in [3]. The data query launched by A circulates through the ad-hoc network and reaches several server-users who have the requested data. Those users will send a positive answer to A. Then, A chooses one of them, denoted by B. Next, A requests the information from B who will send it to A. Finally A sends a receipt to B.

SUs store all the receipts they collect during a given period. At the end of that period, they send all the receipts to the CS in order to get paid according to the number of served requests. Upon getting the receipts, CS checks their validity. If everything is correct, CS pays to the user the money corresponding to the services provided.

2.2 Messages

In our proposal, we distinguish two types of communication:

- The first type follows a client-server paradigm and involves the content source CS. We have chosen this approach because the communication between CS and the other two entities (SU, EU) only occurs at very specific moments and is unlikely to cause a bottleneck.
- The second type of communication, the dialog between an EU and a SU, follows a P2P paradigm.

In both cases, the communication uses structured messages coded with XML. Messages consist of two parts. The first part contains the message itself, divided in two or three sections: message type, sender identifier (in the P2P environment) and message body. The second part contains the cryptographic data: the signature on the first part of the message, the algorithm used to calculate the signature and the digital certificate for the sender's public key. This structure allows the receiver to verify the validity of the message.

2.3 Protocols

In this section, we detail the different protocols used by the entities participating in the system:

- End-user registration
- Server-user registration
- Information request
- Server-user payment

The following notation is used in the rest of the paper:

- P_{entity}, S_{entity}: Asymmetric key pair of *entity*, where P_{entity} is the public key and S_{entity} is the private key.
- $S_{entity}[m]$: Digital signature of message m by *entity*. By digital signature we refer to computing the hash value of message m using a collision-free one-way hash function and encrypting this hash value using the private key of *entity*.

- $E_{entity}(m)$: Encryption of message m under the public key of *entity*.
- $D_{entity}(c)$: Decryption of message c under the private key of *entity*.
- $H(m)$: Hash value of message m using a collision-free one-way hash function.
- $m_1||m_2$: Concatenation of messages m_1 and m_2.

End-user registration. To register as an end-user, a candidate user must contact the CS and install the necessary application software. *E.g.* in our tourist information case study we assume that there exist several places in the city (*e.g.* airport, railway station or tourist office) where a user can register. The end-user registration protocol is as follows:

Protocol 1

1. *The user does:*
   (a) *Obtain the following information from the CS:*
      - *Internet address from which to download the application software.*
      - *Validity period,* i.e. *time window during which the user will be allowed to use the system.*
      - *Access code to download and install the software. In our case study implementation the access code consists of 16 alphanumeric characters, for instance A3GZ-BB44-223G-AGDR.*

   (b) *Connect her device to the Internet and download the application software.*
   (c) *Install the application software.*
   (d) *Run Procedure 1 below and obtain the private key S_{EU} in a PKCS#8 file [12], and a Certificate Signing Request (CSR).*
   (e) *Send the CSR to the Content Provider.*
2. *The CS does:*
   (a) *Issue the user's certificate using the CSR.*
   (b) *Add the issued certificate to the CS database.*
   (c) *Send the issued certificate to the user.*
3. *The user stores the following information in a PKCS#12 [12] file:*
   - *User private key S_{EU}.*
   - *User certificate.*
   - *CS certiticate.*

Procedure 1

1. *Generate a private/public RSA key pair [14].*
2. *Store the private key in a PKCS#8 file.*
3. *Generate a Certificate Signing Request (CSR). The file must use the PKCS# 10 [12] standard.*
4. *Return the PKCS#8 file and the CSR.*

Server-user registration. A user wishing to register as a server-user contacts the CS from whom she will receive a unique identifier and the software that will

enable her to serve information. Afterwards, the user generates a private/public key pair and sends her public key and her identifier to the CS in order to get the corresponding certificate. Finally, the user indicates the desired information items and downloads them to her hard disk. More formally, the server-user registration protocol is as follows:

Protocol 2

1. *The user does:*
   (a) *Sign a contract with the CS specifying the user's rights and duties.*
   (b) *Send the user's bank data for future payments to CS. For confidentiality, these data are sent encrypted under the public key of CS.*
2. *CS does:*
   (a) *Generate a unique identifier* Id.
   (b) *Send to the user the unique identifier* Id *and the software that will enable the user to serve information. For confidentiality,* Id *is sent encrypted under the public key* P_{EU} *(the candidate server-user is assumed to be already an end-user with a private/public key pair* (S_{EU}, P_{EU})*).*
3. *The user does:*
   (a) *Run Procedure 1 to obtain the private key* S_U *in a PKCS#8 file, and a Certificate Signing Request (CSR).*
   (b) *Send the CSR to CS.*
4. *CS does:*
   (a) *Issue the user's certificate using the CSR.*
   (b) *Add the issued certificate to CS's database.*
   (c) *Send the issued certificate to the user.*
   (d) *Send the catalog information.*
5. *The user stores the following information in a PKCS#12 file:*
   – *User private key* S_U.
   – *User certificate.*
   – *CS certificate.*

Information request. When an end-user requests an information item, the query reaches several server-users. Among these, those holding the requested item return a positive acknowledgment. Then, the end-user downloads the requested information from a particular server-user selected among those which have sent positive acknowledgment. Finally, the end-user sends a receipt to the selected server-user. As we will see later on, the SU will use this receipt in order to claim the corresponding reward from the CS.

Now, we describe this protocol in more detail:

Protocol 3

1. *The end-user EU computes a request in order to obtain a specific information, where the request consists of the following data:*

- *Description of the requested item, I.*
- *Date and time of the request, T_r.*
- *Digital signature of I and T_r, $S1 = S_{EU}[I||T_r]$.*

This query spreads using any of the methods described in [3].

2. *Each server-user SU who receives the query does:*
   (a) *Verify the digital signature $S1$ using EU's public key.*
   (b) *Search for the information.*
   (c) *If I is in SU's database, reply to EU. The reply contains the following data:*
      - *User's request, $R_{EU} = I||T_r$.*
      - *Date and time of the answer, T_a.*
      - *Digital signature on R_{EU} and T_a, that is, $S2 = S_{SU}[R_{EU}||T_a]$.*
3. *EU does:*
   (a) *Collect the replies from the SUs. Without loss of generality, assume that the set of SU replying to EU is $SU_1, SU_2, \cdots, SU_n$. See Section 3.3 below on the value of n.*
   (b) *Verify the digital signatures of the SUs, that is, $S2_1, S2_2, S2_3, \cdots, S2_n$ using the public keys of each SU.*
   (c) *Choose one server-user SU' $\in \{SU_1, SU_2, \cdots, SU_n\}$. This choice can be performed in a way to maximize privacy (see Section 3.3 below).*
   (d) *Send a request to SU' with the following data:*
      - *Description of the requested information, I.*
      - *Date and time of the request, T_r.*
      - *Identifier of the node this request is addressed to, $Id_{SU'}$.*
      - *Digital signature on I, T_r and $Id_{SU'}$, that is, $S3 = S_{EU}[I||T_r||Id_{SU'}]$.*
4. *SU' does:*
   (a) *Verify the digital signature $S3$ using the public key P_{EU}.*
   (b) *Send the following message:*
      - *Description of the requested information, I.*
      - *Requested information, $Info$.*
      - *Date and time of the answer, T_a.*
      - *Digital signature of the I, $Info$ and T_a, that is, $S4 = S_{SU'}[I||Info||T_a]$.*
5. *EU does:*
   (a) *Verify the digital signature $S4$ using P_{SU}.*
   (b) *Check whether the received data correspond to the information requested.*
   (c) *If the check is OK, issue a receipt and send it to SU' with the following data:*
      - *Description where EU asserts that she has received the item described as I from SU'.*
      - *Date and time, T.*
      - *Identifier of SU', $Id_{SU'}$.*
      - *Digital signature on I, T, and Id_{SU}, that is, $S5 = S_{EU}[I||T||Id_{SU}]$.*
6. *SU' does:*
   - *Receive the receipt.*
   - *Verify $S5$ using P_{EU}*
   - *Store the receipt.*

Server-user payment. As previously described, server-users get a receipt every time they serve information. These receipts are stored. Once a large enough batch of receipts has been collected, a server-user contacts CS to get paid for the services provided. Note that sending receipts one at a time to CS would be very inefficient. The reason is that, since the reward for a single service is very low, the processing costs of such a payment would be too significant.

The protocol to redeem a batch of receipts is as follows:

Protocol 4

1. *SU sends the receipts to CS.*
2. *CS does:*
   (a) *Verify the digital signature of each receipt*
   (b) *Check for duplicated receipts*
   (c) *Compute the money that must be paid to the information node*
   (d) *Transfer the money to the bank account of SU*

3 Security and Privacy Analysis

Our communication protocols use different types of messages to be transmitted in each phase. Every exchanged message contains a plaintext part and a valid signature. The plaintext part contains the information transmitted between nodes and the signature provides *authentication*, *integrity* and *non-repudiation* to such messages.

3.1 Confidentiality

In principle, confidentiality is only implemented in the server-user registration protocol, when the user sends her bank data to CS and CS returns a unique identifier (Steps 1 and 2 of Protocol 2). The rest of messages are assumed to be non-confidential, which is plausible for most applications (*e.g.* tourist information). However, if confidentiality is required, it can be achieved by encrypting messages under the public key of the intended receiver.

3.2 Collusion Security

Collusion between end-users and server-users to obtain unlawful rewards is conceivable: some end-users perform a huge amount of information requests to certain server-users, and the latter then share with the former the rewards obtained from the CS.

A possible solution is to charge the end-users a small fee for enjoying the information service. This payment can be performed using offline electronic checks as stated in [2] or any micropayment system (*e.g.* PayWord, [15]).

However, one must acknowledge that collecting payment from the end-users can jeopardize the success of many applications, like the tourist information system. Therefore, a preferred countermeasure against user collusion is for the

CS to record and analyze the number of receipts submitted by the SUs and the number of receipts issued by the same EU. Since each receipt contains the exact time and date when it was issued, a limit on the number of requests that an EU can perform within a period of time can easily be enforced. The CS will not honor any receipts beyond those that can be issued by a certain user; furthermore, as soon as CS detects that an end-user has issued more receipts than allowed, CS alerts the SUs to stop serving any further request from that suspect SU. The SUs receive this alert when they synchronize resources with the CS or when they redeem their receipts. In this way, the effects of possible user collusions are tolerably mitigated.

3.3 Privacy

In any information service, end-user profiling is a real threat. Indeed, information providers can keep track of the requests submitted by end-users, with a view to investigating their tastes, preferences, locations, etc. This is clearly a potential privacy violation.

In a conventional information service where end-users get information directly from a single information provider, one often assumes that information provider to be trusted or at least not to be interested in violating the privacy of end-users. At any rate, if there ever were any provable violation, the information provider would be liable and could be charged accordingly.

In a peer-to-peer mobile ad-hoc information service, the privacy problem is much more serious. End-users obtain information through server-users who are occasional information relayers and cannot be trusted to the same extent as to privacy preservation.

End-user privacy can be significantly increased by using an alias when registering as an end-user and by properly tuning Protocol 3:

- When the end-user application detects that there are server-users among the n replying to Step 3a who already replied to more than p requests from the same end-user in the past (p is a privacy parameter), the application warns the end-user of a potential privacy problem. The end-user has two choices: either move to a different area where she will find different server-users or to go ahead and jeopardize her privacy.
- In Step 3c, a wise policy is for the end-user application to choose the server-user which has replied to least requests to the end-user in the past.

Of course, we are assuming that the server-user application has not been tampered with, so that: i) it replies when the server-user hears a request for an information item it holds; ii) it forgets about requests for information items the server-user does not hold.

In the presence of malicious server-users, a combination of the following two strategies can be useful:

- Use short validity periods for end-users, which will force end-users to frequently re-register under a new alias.

– Avoid issuing many information request from the same place, which should be easy for a roaming end-user (*e.g.* tourist visiting a city). Moving to another area is a way to get rid from the current server-users, both the legitimate and the malicious ones.

4 Implementation

We have implemented a functional prototype that demonstrates the main capabilities of the system previously described. Although not all funtionalities are currently implemented as of this writing, the system core is fully operational already.

We have chosen Java [7] as a language for this implementation, due to its portability to different architectures that need to interoperate, an essential feature in our scenario. We have used JXTA [8] to establish the peer-to-peer communications, although there are other options like Gnutella or Bamboo. Finally, to implement the client-server paradigm (for communication with the CS), we use the remote method invocation protocol (RMI) associated to the Java technology.

Specifically, the content source and the server-users have been implemented with Java SE version 5.0 and they use databases containing touristic resources that run on MySQL version 5.0. The CS component executes a secure web server, which acts as an access point for new server-users willing to join the system. In the future we will allow the end-users to enter the system through the secure web server too. SU applications make use of JXTASE libraries, version 2.4, to communicate with the other entities through the peer-to-peer ad-hoc network.

The end-user application is implemented in a WiFi enabled HP IPAQ series hx2700. We have installed the J9 Java Virtual Machine from IBM [9] that works with Connected Device Configuration 1.1 and is PersonalJava 1.2 compliant. This PersonalJava version follows the Java 1.1.8 version specifications. The JXTA version that works on the PDA is the JXME 2.1.3.

4.1 Server-User Registration

Once a user runs the SU application for the first time (Protocol 2), she must introduce an alias and a password of her choice, and enter the CS URL as well. After this process, she generates a key pair RSA (1024 bits) and a Certificate Signing Request (CSR) to be sent to the CS. This entity sends back the CSR properly certified. Finally, the user generates the PKCS#12 file and a configuration file, which will be used in the following executions and can be modified any moment from the SU graphical interface. This configuration file contains the user's alias, the URL of the CS and the absolute path to the PKCS#12 file which contains the user's private key, the user's certificate and the certificate of the CS.

After this initial phase, the SU application displays a first window with a menu showing some options to be chosen by the user. Then, she can pick the items from a catalog provided by the CS. The selected items will be stored in

the user's computer and will be used to provide information to the end-users. After serving some information requests from end-users, the SU can check the number of receipts stored and the money earned. She can decide to redeem the stored receipts.

4.2 End-User Registration

We assume that, in Step 1 of Protocol 1, the candidate end-user receives her registration information (Internet address from which to download, validity period, access code) offline, *e.g.* in a tourist office. After this (remaining steps of Protocol 1), the user goes on-line and, much like a server-user, the end-user submits her Certificate Signing Request to CS, who will return it properly certified. At the end of this transaction, the end-user will obtain a PKCS#12 file and a configuration file will be created. Upon completion of this installation process, the end-user gets her keys and is ready to use the information service.

In the tourist information service example, once the user reaches the vicinity of a monument, she may be interested in obtaining some information about it. For that, she must chose the corresponding item in the catalog, using the EU application in PDA and send the request through the ad-hoc peer-to-peer network to SU applications present in the area. In a few seconds, a list of SU nodes offering the requested item will appear on the end-user's screen. The user then chooses one of the possible providers and the selected node sends the desired information.

5 Conclusions and Future Work

An architecture for a peer-to-peer mobile ad-hoc network offering distributed information provision has been presented. The novelty of our proposal lies on the incentive scheme to encourage users to become distributed information servers. Our implementation demonstrates the feasibility and the usefulness of our approach.

The most challenging part of our system is security and, especially, privacy. Indeed, allowing any user to behave as a server is not without risks. While authentication, integrity, non-repudiation and even confidentiality can be guaranteed to a large extent, preserving the privacy of end-users is trickier because they can be profiled by server-users. We have proposed and implemented some basic privacy-preserving countermeasures, but we feel that privacy in peer-to-peer ad-hoc networks is a far-reaching problem that definitely deserves further research. In the specific case of our system, a challenging issue is to design privacy countermeasures which are minimally inconvenient for the end-user, *e.g.* that do not require very short validity periods or constant roaming.

Disclaimer and Acknowledgments

The authors are solely responsible for the views expressed in this paper, which do not necessarily reflect the position of UNESCO nor commit that organization.

This work was partly supported by the Spanish Ministry of Education through project SEG2004-04352-C04-01 "PROPRIETAS" and by the Government of Catalonia under grant 2005 SGR 00446.

References

1. Blundo, C., Cimato, S., De Bonis, A.: Secure E-Coupons. Electronic Commerce Research Journal 5(1), 117–139 (2005)
2. Chaum, D., Den Boer, B., Van Heyst, E., Mjolsnes, S., Steenbeek, A.: Efficient offline electronic checks (extended abstract). In: Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques on Advances in Cryptology. Springer, New York (November 1990)
3. Ding, G., Bhargava, B.: Peer-to-peer file-sharing over mobile ad hoc networks. In: First International Workshop on Mobile Peer-to-Peer Computing, p. 1000 (2004)
4. Greer, J., McCalla, G., Vassileva, J., Deters, R., Bull, S., Kettel, L.: Lessons learned in deploying a multi-agent learning support system: the I-help experience. In: Proceedings of AI in Education AIED'2001, pp. 410–421 (2001)
5. Heinemann, A., Kangasharju, J., Lyardet, F., Mühlhäuser, M.: iClouds - Peer-to-peer information sharing in mobile environments. In: Kosch, H., Böszörményi, L., Hellwagner, H. (eds.) Euro-Par 2003. LNCS, vol. 2790, pp. 1038–1045. Springer, Heidelberg (2003)
6. Heinemann, A., Mühlhäuser, M.: Spontaneous collaboration in mobile peer-to-peer networks. In: Steinmetz, R., Wehrle, K. (eds.) Peer-to-Peer Systems and Applications. LNCS, vol. 3485, pp. 419–433. Springer, Heidelberg (2005)
7. Sun Microsystems, JAVA Programming language: http://java.sun.com
8. Sun Microsystems, Project JXTA: http://www.jxta.org/
9. IBM, WebSphere Everyplace Micro Environment: http://www-128.ibm.com/
10. Li, J., Kang, X.: Proof of Service in a Hybrid P2P Environment. In: Chen, G., Pan, Y., Guo, M., Lu, J. (eds.) Parallel and Distributed Processing and Applications - ISPA 2005 Workshops. LNCS, vol. 3759, pp. 64–73. Springer, Heidelberg (2005)
11. Pan, J., Cai, L., Shen, X., Mark, J.W.: Identity-based secure collaboration in wireless ad hoc networks. Computer Networks 51(3), 853–865
12. Public-Key Cryptography Standards (PKCS): http://www.rsasecurity.com/rsalabs/node.asp?id=2124
13. Rajasekaran, H.: An incentive based distribution system for DRM protected content using peer-to-peer networks. In: 1st International Conference on Automated Production of Cross Media Content for Multi-channel Distribution, pp. 150–156 (2005)
14. Rivest, R.L., Shamir, A., Adleman, L.M.: A method of Obtaining Digital Signatures and Public-Key Cryptosystems. Communications of the ACM 21(2), 120–126 (1978)
15. Rivest, R.L., Shamir, A.: PayWord and MicroMint: two simple micropayment schemes. In: Lomas, M. (ed.) Security Protocols. LNCS, vol. 1189, pp. 69–87. Springer, Heidelberg (1997)
16. Vassileva, J.: Motivating Participation in Peer to Peer Communities. In: Petta, P., Tolksdorf, R., Zambonelli, F. (eds.) ESAW 2002. LNCS (LNAI), vol. 2577, pp. 141–155. Springer, Heidelberg (2003)
17. Vishnumurthy, V., Chandrakumar, S., Sirer, E.: Karma: A secure economic framework for p2p resource sharing. In: Workshop on Economics of Peer-to-Peer Systems (2003)

Classification of Normal and Tumor Tissues Using Geometric Representation of Gene Expression Microarray Data

Saejoon Kim and Donghyuk Shin

Department of Computer Science
Sogang University
Seoul 121-742, Korea
{saejoon, shindh}@sogang.ac.kr

Abstract. Microarray is a fascinating technology that provides us with accurate predictions of the state of biological tissue samples simply based on the expression levels of genes available from it. Of particular interest in the use of microarray technology is the classification of normal and tumor tissues which is vital for accurate diagnosis of the disease of interest. In this paper, we shall make use of *geometric representation* from graph theory for the classification of normal and tumor tissues of colon and ovary. The accuracy of our geometric representation-based classification algorithm will be shown to be comparable to that of the currently known best classification algorithms for the two datasets. In particular, the presented algorithm will be shown to have the highest classification accuracy when the number of genes used for classification is small.

1 Introduction

Microarray technology is a very powerful and popular tool nowadays capable of providing high throughput simultaneous measurement of expression levels of thousands of genes which can accurately represent the state of a biological cell or tissue of interest. For this reason, it is used for many applications including identification of various types of disease-related genes [9] [10] [16], classification of biological samples into normal and diseased [2] [12] [21], or two or more subcategories of a given disease [1] [4] [9] [15] [20] [23] [24], pharmacogenomics and clinical studies [7] [13] [14], and many more.

The classification of normal and cancerous tissues, or cancer into its subtypes is an important application area of microarray technology since correct diagnosis is vital for successful treatment of it. For this reason, a variety of classification methods has been investigated in the literature in the past including the initial work of [9] that used a simple voting scheme to classify the human acute leukemia into acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML). Since then various types of cancer have been analyzed by numerous

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 393–402, 2007.

classification methods including the works of [1], [15], and [23]. In [1] and [23], diffuse large B-cell lymphoma (DLBCL) and 60 cell lines from the National Cancer Institute's anti-cancer drug screen (NCI 60), respectively, were classified into their subcategories using hierarchical clustering, and in [15], small round blue cell tumors (SRBCTs) were classified using neural networks. In [20] and [24], breast cancer was classified, and in [2] and [4], colon cancer was classified. Somewhat extensive comparisons of the numerous classification methods on various datasets that have appeared in the literature have been presented in [6], [17], and [22].

Currently widely used methods for classification or class prediction of cancer that have high classification accuracy are schemes based on support vector machines (SVMs) [8] [19], and ones based on linear discrminant analysis (LDA) [5] [25]. SVMs are a kernel-based learning algorithm that have foundations on the structural risk minimization principle in statistical learning theory. SVMs perform quite well and currently have the highest classification accuracies on various datasets. LDA is a classical method for classification that have spawned a variety of generalizations. LDA-based algorithms also perform quite well and possess the advantage of easy interpretability.

In this paper, we present a fundamentally different approach from the techniques that have been used to solve classification of gene expression data of cancer. Here, we formulate the problem of classification of biological samples as that of finding a min-cut of a graph representation of the samples, where the graph representation accurately reflects the similarities and dissimilarities of all pairs of samples by using edges and no edges between pairs, respectively. One can deduce from the results in [9] that in this graph representation, samples of the same class are likely to be highly connected to each other while samples of different classes are not if the similarity measure used to generated the graph representation is not too different from the one used in [9]. Thus finding a min-cut of the graph representation can establish a classification of samples into their respective classes. To this end, we present a heuristic from graph theory called *spectral bipartitioning* [3] [11] [18] that computes a geometric representation of the graph from which a min-cut of the graph can be derived.

The contribution of this paper is as follows. We first show how to generate a *similarity graph* of samples that accurately represents the similarities and dissimilarities between all pairs of samples. Then we apply a spectral bipartitioning heuristic to the similarity graph to obtain a geometric representation of it which is essentially a classification of the samples. It will be shown that the proposed geometric representation-based classification algorithm has accuracy that is comparable to that of some of the currently known best classification methods for two publicly available datasets of colon cancer of [2] and ovarian cancer of [21]. In particular, the proposed classification algorithm in this paper will be shown to have the highest accuracy when the number of genes used for classification is small, say, less than 10.

2 Systems and Methods

2.1 Preliminaries

Gene expression levels from microarray are usually represented by the expression profile matrix P of m rows and n columns where m and n are the number of genes and samples, respectively. So, in $P = (p_{ij})$, $p_{ij} \in \mathbb{R}$, $1 \leq i \leq m$, $1 \leq j \leq n$, each row corresponds to expression profile vector of a gene and each column corresponds to that of a sample, and p_{ij} is the expression level of gene i in sample j. Denote the expression profile vector of ith gene by $g_i(\stackrel{\triangle}{=} (p_{i1}, \cdots, p_{in}))$ and that of jth sample by $s_j(\stackrel{\triangle}{=} (p_{1j}, \cdots, p_{mj}))$.

For each expression profile vector of a sample or simply a *sample vector*, there is its associated class label which characterizes the biological state of the sample. For example, the ith sample vector s_i has its class label $y_i \in \{0, 1\}$ if the number of biological states or classes is two. The classification problem of interest in this paper involves the class assignment to a vector of unknown binary class label given sample vectors with known binary class labels. More formally, we want to build a classification algorithm that can accurately predict the class label of a vector given the following two sets

$$S = \{s_1, \cdots, s_n\} \subseteq \mathbb{R}^m$$
$$Y = \{y_1, \cdots, y_n\} \subseteq \{0, 1\}$$

where s_i has class label y_i, $i = 1, 2, \cdots, n$. This set S with its associated set of class labels Y is called the training set, and set of vectors of unknown class labels whose values are to be determined is called the test set. Hence a classification algorithm works in two steps: the *training step* during which a class assignment rule is established from the training set, and the *testing step* during which this rule is applied to the test set to obtain the prediction of its vectors' class labels.

The accuracy of the classification algorithm is tested by performing the leave-one-out cross validation (LOOCV) in this paper. In other words, of the sample vectors provided with class labels, we will use all but one of them, *i.e.*, $n - 1$ sample vectors, as the training set and the remaining one sample vector as the test set to check the accuracy of the classification scheme. We repeat this process until all of the sample vectors have been used once as the test set after which the LOOCV process is completed. Since the class label of each of the test set is known, the accuracy of the classification scheme can be calculated by comparing the predicted class label with the true label.

The classifier in this paper that predicts the class label of the sample vector in the test set will utilize the similarities or the dissimilarities between the sample vectors in the training set. To this end, we shall use in this paper a widely adopted similarity measure in the literature. It is the Euclidean distance measure and is defined as

$$\mathsf{d}(s_i, s_j) = \sqrt{\sum_{k=1}^{m} (p_{ki} - p_{kj})^2}$$

for sample vectors s_i and s_j. In this similarity measure, smaller value of $\mathsf{d}(s_i, s_j)$ represents higher similarity or closeness between the two sample vectors s_i and s_j.

To create a classification algorithm that has high accuracy, it is normally better to use only a subset of the genes that have high discriminatory power [6] [9] instead of all the genes as can be noted in the definition of d. This selection of a subset of genes for classification is more generally called *feature selection* in the computer science community. It is important to note that feature selection is established using only the training set data and not in combination with the test data. For example, in performing LOOCV, feature selection is established for each training set every time it is modified. We shall make use of two feature selection schemes in this paper where both schemes use a selected number of genes with highest feature scores for classification. The first scheme was originally used in [9] and its feature score for gene g_i, $f_1(g_i)$, is defined as

$$f_1(g_i) = \left| \frac{\mu_0(g_i) - \mu_1(g_i)}{\sigma_0(g_i) + \sigma_1(g_i)} \right|$$

where $\mu_k(g_i)$ is the mean of gene g_i across samples with class label k and $\sigma_k(g_i)$ is the standard deviation of gene g_i across samples with class label k, $k = 0, 1$. The second scheme was originally used in [6] and defines its feature score for gene g_i, $f_2(g_i)$, as the ratio of gene's between-groups sum of squares to within-groups sum of squares, *i.e.*,

$$f_2(g_i) = \frac{\sum_{j=1}^{n} \sum_{k=0}^{1} 1(y_j = k)(\mu_k(g_i) - \mu(g_i))^2}{\sum_{j=1}^{n} \sum_{k=0}^{1} 1(y_j = k)(p_{ij} - \mu_k(g_i))^2}$$

where the function $1(\cdot)$ equals 1 if the argument if true and equals 0 otherwise, $\mu(g_i)$ is the mean of gene g_i across all samples and $\mu_k(g_i)$ is the same as above in the definition of $f_1(g_i)$, $k = 0, 1$. Henceforth, the similarity measure d previously defined is modified by replacing the summation in its expression by a partial summation over the indices for which associated genes have high feature scores for each of the two feature selection schemes. We chose 6, 10, and 20 as the selected numbers of genes with highest feature scores used for classification.

2.2 Graph Construction

A graph G is an ordered pair of sets (V, E) where V is the set of vertices and E is the set of edges that are unordered pairs of V. The classification algorithm that we propose in this paper begins with a construction of a graph whose vertices represent the sample vectors. An important feature of this graph is that it shows the similarities between the sample vectors by having two vertices connected by an edge if the associated sample vectors are sufficiently similar to each other, and not having two vertices connected by an edge if otherwise.

Let us now construct the graph that summarizes the similarities between the sample vectors which we call the *similarity graph*. Without loss of generality,

consider the training step of our algorithm. Since we already know that each vertex represents one sample vector, it remains to show how the $n-1$ vertices are connected to each other to finish the graph construction. Note that there can be at most $\binom{n-1}{2}$ edges in the graph. Observe that for each $k \in \{1, \cdots, n-2\}$, there are k closest sample vectors for each sample vector with respect to the Euclidean distance similarity measure. For each $k \in \{1, \cdots, n-2\}$ and feature selection scheme, we complete graph construction by drawing edges between pairs of vertices in the graph if both of the associated sample vectors are among the k closest sample vectors of each other. This ensures that two adjacent vertices in the graph accurately represent "closeness" of the two associated sample vectors. Let us, by abuse of terminolgy, speak of a *similarity graph with maximum vertex degree* k if the graph is generated by connecting each pair of vertices whose associated sample vectors are among the k closest sample vectors of each other. Note that a similarity graph with maximum vertex degree k need not necessarily have a vertex whose degree is k. We will define the *parameter* of a similarity graph by k as in the similarity graph with maximum vertex degree k. Jumping ahead, the class assignment rule to be established during the training step of our classification algorithm is finding the parameter value that generates "good" bipartitioning of the training set.

Let us illustrate a similarity graph construction by using a toy example in which there are four sample vectors $s_1, \cdots, s_4$, and let $N_k(s)$ denote the set of k closest sample vectors of sample vector s. Suppose that

$$\begin{aligned} N_2(s_1) &= \{s_2, s_3\}, \\ N_2(s_2) &= \{s_1, s_4\}, \\ N_2(s_3) &= \{s_2, s_4\}, \\ N_2(s_4) &= \{s_2, s_3\}. \end{aligned}$$

Observe that s_1 and s_2 are both among the 2 closest sample vectors of each other while s_1 and s_3 are not. Therefore, if we denote by $G_k = (V_k, E_k)$ the generated similarity graph with maximum vertex degree k, we have

$$\begin{aligned} V_2 &= \{v_1, v_2, v_3, v_4\}, \\ E_2 &= \{(v_1, v_2), (v_2, v_4), (v_3, v_4)\} \end{aligned}$$

where v_i corresponds to sample vector s_i, $i = 1, \cdots, 4$.

2.3 Geometric Representation

This subsection describes how a geometric representation or a linear ordering of the similarity graph of sample vectors is computed. This geometric representation is obtained from a spectral bipartitioning heuristic and induces a bipartition or a classification of samples into two classes. Hence, geometric representation, bipartition, and classification all essentially have the same meaning. *Spectral bipartitioning* heuristic is a widely used method in VLSI and various scientific computing communities for partitioning a graph into two clusters so that the

number of edges cut between the two clusters is minimized. It relates the eigenvector associated with the second smallest eigenvalue of the Laplacian matrix of the graph to bipartitioning. An objective in this bipartitioning involves some bounds on the cluster sizes because solutions to the min-cut of a graph generally tend to produce clusters of largely different sizes which will not be of much practical value. For example, if we consider a graph that has a vertex with a very small degree, a cut to the edges of this vertex can be a min-cut which clearly is not a meaningful bipartitioning. For this reason, various versions of spectral bipartitioning heuristic exist in the literature [3] [11] [18] that aim at producing more practical solutions to the bipartitioning problem. One practical solution to the bipartitioning problem tries to minimize the *ratio cut* of a graph $G = (V, E)$ which is defined as

$$\frac{|e(W, V \backslash W)|}{|W||V \backslash W|}$$

where W is a subset of V, $V \backslash W$ is the complement of W in V, $e(W, V \backslash W)$ is the set of edges between two sets of vertices W and $V \backslash W$, and $|\cdot|$ is the cardinality of the set $\cdot$. (Note that if there are no bounds on the cluster sizes, then the solution to the bipartitioning problem only minimizes the numerator in the expression for the ratio cut.) This minimization of the ratio cut of a graph captures the essence of a min-cut while favoring an even bipartition [11]. We shall therefore use the spectral bipartitioning heuristic whose goal is minimizing this ratio cut in this paper.

To describe the spectral bipartitioning heuristic, let us introduce some definitions from graph theory. The adjacency matrix $A = (a_{ij})$ of a graph with n vertices $v_1, v_2, \cdots, v_n$ is the symmetric $n \times n$ matrix where $a_{ij} = 1$ if there is an edge between v_i and v_j, and $a_{ij} = 0$ otherwise. Note that the diagonals of A are all zeros. The diagonal matrix $D = (d_{ij})$ of the graph is defined by $d_{ii} = deg(v_i)$ and $d_{ij} = 0$ for $i \neq j$ where $deg(v_i)$ is the degree of the vertex v_i. The Laplacian matrix L of the graph is defined as $L = D - A$. So the elements in each row or column of a Laplacian matrix sum to zero. ¿From the semi-positive definiteness of Laplacian matrices, Laplacian matrix has n eigenvalues that are real and nonnegative, *i.e.,* $0 \leq \lambda_1 \leq \cdots \leq \lambda_n$. The following, which is adopted from [11], is the spectral bipartitioning heuristic that minimizes the ratio cut.

Spectral Bipartitioning

1. *Compute the eigenvector μ_2 corresponding to the second smallest eigenvalue λ_2 of the Laplacian matrix L of graph G.*
2. *For each vertex v_i in G, if the ith coordinate of μ_2 is less than* 0*, then assign vertex v_i with class label* 0*. Otherwise, assign vertex v_i with class label* 1*.*

The geometric representation of the graph G is induced by the eigenvector μ_2 and is a sequential list of vertices in which vertices are ordered according to their respective values in the coordinates of μ_2. Geometric representation thus in turn induce a bipartition or a classification of the graph G. "Spectral Bipartitioning" divides a connnected graph G into two components G_0 and G_1

such that there are no edges between the two components G_0 and G_1, and each component is a connected graph, or a cluster. We now describe our *Geometric Representation-based Classification (GRbC)* algorithm.

Geometric Representation-based Classification

1. *Generate similiarity graphs of training set each of which has $n-1$ vertices.*
2. *For the training step, apply "Spectral Bipartitioning" to each similarity graph of training set and identify the parameter of graph that yields "best-effort" bipartitioning.*
3. *For the testing step, use this parameter to create the similarity graph of sample vectors in both training and test sets.*
4. *"Spectral Bipartitioning" is applied to this graph with n vertices after which the class label of the sample vector in the test set is assigned to the majority of class labels of other sample vectors in the same cluster.*

"Best-effort" bipartitioning here means the bipartitioning that has the most number of vertices of the same class label in the same cluster for each cluster. In the ideal case of "perfect" bipartitioning, all vertices of the same class label are in the same cluster for each class label. The last step in "GRbC" needs a little explanation since assigning the sample vector in the test set to the majority of class labels of other sample vectors in the same cluster may not seem like the best choice of classification when the numbers of sample vectors in the two classes are nontrivially disproportioned. In other words, the last step in "GRbC" may seem a biased assignment towards the class label with more sample vectors. For this matter, we also tried a different class assignment rule that offsets this bias and makes a "fair" judgement. It turns out that for parameter values that return best-effort bipartitioning, the two assignment rules had almost the same classification result which justifies this last step in "GRbC." We also note that there can be multiple parameter values that yield the same best-effort bipartitioning and a particular choice of parameter value had little effect on classification accuracy.

3 Results

In this section, we show the classification results of our proposed "GRbC" and four of the currently known best classification algorithms for microarray data applied to the mentioned two datasets of cancer. Table 1 shows the LOOCV classification accuracies for the colon dataset which has 62 sample vectors. In this dataset, there are 40 tumor sample vectors and 22 normal ones. The values in the table correspond to those of higher classification accuracy from either of the two feature selection schemes. (The classification accuracies of the two feature selection schemes were very close and neither one was consistently better than the other for all of the datasets considered in this paper.) The first row in the table shows the result of our "GRbC," the second row shows that of SVM using a linear kernel function, and the third row shows the same using a radial basis kernel function. (In Tables 1 and 2, we tried out enough values for the variables in

SVMs to optimize performance.) The forth row shows the classification accuracy of PAM of [25] and the last row shows that of ClaNC of [5]. The table clearly indicates that our "GRbC" is comparable to other classification algorithms listed in the table, and in particular, "GRbC" has the highest classification accuracy when the number of genes used is 6 and 10.

Table 1. Comparison of Classification Accuracies for Colon Dataset

	number of genes		
	6	10	20
GRbC	87.1%	87.1%	87.1%
SVM (linear)	85.5%	85.5%	90.3%
SVM (radial)	83.9%	85.5%	85.5%
PAM	82.4%	87.1%	87.1%
ClaNC	85%	87%	89%

Table 2 shows the LOOCV classification accuracies for the ovarian dataset which has 253 sample vectors. In this dataset, there are 162 tumor sample vectors and 91 normal ones. The results shown in the table are similar to those of the previous table, and in particular, "GRbC" has the highest classification accuracy when the number of genes used is 6.

Table 2. Comparison of Classification Accuracies for Ovarian Dataset

	number of genes		
	6	10	20
GRbC	97.6%	98%	98.4%
SVM (linear)	96.8%	98.4%	94%
SVM (radial)	96.4%	98.8%	98.4%
PAM	96.8%	98.8%	98.8%
ClaNC	97%	98%	98%

To sum up, in comparison with four of the best classification algorithms known for microarray data, "GRbC" had the highest classification accuracy in all datasets considered in this paper when number of genes used for classification was 6. Moreover, in one dataset considered, "GRbC" had the highest classification accuracy when numbers of genes used were 6 and 10.

4 Discussion

In this paper, a classification technique based on a geometric representation of graph was presented. The technique exhibits a fundamentally different approach from the ones that have been known to classify gene expression data of cancer.

In this technique, the classification problem was formulated as a problem of finding a min-cut of similarity graph induced by the gene expression dataset which was solved through a spectral bipartitioning heuristic. It turns out that this classification technique has accuracy that is comparable to some of the currently known best classification algorithms for the three datasets considered in this paper. This classification algorithm, in particular, showed the highest classification accuracy for all of the datasets considered when the number of genes used was small. Furthermore, the algorithm facilitates easy interpretability due to its conceptual simplicity.

Acknowledgement

We are thankful to Alan Dabney for sharing the datasets used in his paper and for the preprints of his work. This research work was supported in part by Sogang University Grant 20061022.

References

1. Alizadeh, A., et al.: Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature 403, 503–511 (2000)
2. Alon, U., et al.: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. In: Proc. Nat'l. Acad. Sci. USA, vol. 96, pp. 6745–6750 (1999)
3. Barnes, E.R.: An algorithm for partitioning the nodes of a graph. SIAM Journal of Algebraic and Discrete Methods 3, 541–550 (1982)
4. Barrier, A., et al.: Colon cancer prognosis prediction by gene expression profiling. Oncogene 24, 6155–6164 (2005)
5. Dabney, A.: Classification of microarrays to nearest centroids. Bioinformatics 21, 4148–4154 (2005)
6. Dudoit, S., et al.: Comparison of discriminant methods for the classification of tumors using gene expression data. University of California, Berkeley, Technical Report #576 (2002)
7. Ernst, J., Bar-Joseph, Z.: STEM: a tool for the analysis of short time series gene expression data. BMC Bioinformatics 7(191) (2006)
8. Furey, T., et al.: Support vector machine classification and validation of cancer tissue samples using microarray expression data. Bioinformatics 16, 904–914 (2000)
9. Golub, T., et al.: Molecular classification of cancers; class discovery and class prediction by gene expression monitoring. Science 286, 531–537 (1999)
10. Guyon, I., et al.: Gene selection for cancer classification using SVM. Mach. Learning 46, 389–422 (2002)
11. Hagen, L., Kahng, A.B.: New spectral methods for ratio cut partitioning and clustering. IEEE Trans. Computer-Aided Design 11, 1074–1085 (1992)
12. Iizuka, N., et al.: Oligonucleotide microarray for prediction of early intrahepatic recurrence of hepatocellular carcinoma after curative resection. The Lancet 361, 923–929 (2003)
13. Kaminski, N., Bar-Joseph, Z.: A patient-gene model for temporal expression profiles in clinical studies. In: Apostolico, A., Guerra, C., Istrail, S., Pevzner, P., Waterman, M. (eds.) RECOMB 2006. LNCS (LNBI), vol. 3909, pp. 69–82. Springer, Heidelberg (2006)

14. Kato, T., et al.: Network-based de-noising improves prediction from microarray data. BMC Bioinformatics 7(S4) (2006)
15. Khan, J., et al.: Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nat. Med. 7, 673–679 (2001)
16. Kluger, Y., et al.: Spectral biclustering of microarray data: co-clustering genes and conditions. Genome Res. 13, 703–716 (2003)
17. Lee, J., et al.: An extensive comparison of recent classification tools applied to microarray data. Comput. Stat. Data Anal. 48, 869–885 (2005)
18. Mohar, M.: The Laplacian spectrum of graphs. In: Graph Theory, Combinatorics, and Applications, pp. 871–898. Wiley, Chichester (1991)
19. Mukherjee, S., et al.: Support Vector Machine Classfication of Microarray Data. MIT A.I. Memo No. 1677 and C.B.C.L. Paper No. 182 (1998)
20. Perou, C., et al.: Molecular portraits of human breast tumours. Nature 406, 747–752 (2000)
21. Petricoin III, E., et al.: Use of proteomic patterns in serum to identify ovarian cancer. The Lancet 359, 572–577 (2002)
22. Rajagopalan, D.: A comparison of statistical methods for analysis of high density oligonucleotide array data. Bioinformatics 19, 1469–1476 (2003)
23. Ross, D., et al.: Systematic variation in gene expression patterns in human cancer cell lines. Nat. Genet. 24, 227–235 (2000)
24. Sotirou, C., et al.: Breast cancer classification and prognosis based on gene expression profiles from a population-based study. In: Proc. Nat'l. Acad. Sci. USA, vol. 100, pp. 10393–10398 (2003)
25. Tibshirani, R., et al.: Diagonosis of multiple cancer types by shrunken centroids of gene expression. In: Proc. Nat'l. Acad. Sci. USA, vol. 99, pp. 6567–6572 (2002)

A Seed-Based Method for Predicting Common Secondary Structures in Unaligned RNA Sequences

Xiaoyong Fang[1], Zhigang Luo[1], Zhenghua Wang[1], Bo Yuan[2], and Jinlong Shi[1]

[1] National Laboratory for Parallel & Distributed Processing,
National University of Defense Technology
410073 Changsha, China
zgluo@nudt.edu.cn
[2] Department of Biomedical Informatics, College of Medicine and Public Health,
Ohio State University
43210-1239 Columbus Ohio, USA
yuan.33@osu.edu

Abstract. The prediction of RNA secondary structure can be facilitated by incorporating with comparative analysis of homologous sequences. However, most of existing comparative approaches are vulnerable to alignment errors. Here we use unaligned sequences to devise a seed-based method for predicting RNA secondary structures. The central idea of our method can be described by three major steps: 1) to detect all possible stems in each sequence using the so-called position matrix, which indicates the paired or unpaired information for each position in the sequence; 2) to select the seeds for RNA folding by finding and assessing the conserved stems across all sequences; 3) to predict RNA secondary structures on the basis of the seeds. We tested our method on data sets composed of RNA sequences with known secondary structures. Our method has average accuracy (measured as sensitivity) 69.93% for singe sequence tests, 72.97% for two-sequence tests, and 79.27% for three-sequence tests. The results show that our method can predict RNA secondary structure with a higher accuracy than Mfold.

Keywords: RNA secondary structure, seed-based, unaligned RNA sequences.

1 Introduction

RNA, once considered a passive carrier of genetic information, is now known to play a more active role in nature. It catalyzes reactions, directs the site-specific modification of RNA nucleotides, modulates protein expression and serves in protein localization. Non-coding RNA (ncRNA) genes produce functional RNA molecules rather than encoding proteins. Recent research shows that ncRNA is far more widespread than was previously anticipated [1-2]. The function of an RNA molecule derives from its (secondary) structure. RNA secondary structure, the sum of canonical base pairs, is generally more stable than tertiary contacts and can largely be determined independently of tertiary structure. To date, experiments constitute the most reliable method for secondary structure determination [3]. Unfortunately, their difficulty and expense are often prohibitive. For this reason, computational prediction

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 403–413, 2007.

provides an attractive alternative to discovery of RNA secondary structure. Comparative methods and Minimum Free Energy (MFE) methods [4-5] have been used to predict RNA secondary structures. However, there are several independent reasons why the accuracy of MFE structure prediction is limited in practice [6]. Generally the better accuracy can be achieved by employing comparative methods, in which a large number of sequences are aligned to reveal the common base pairing pattern. So far a number of methods based on comparative analysis of homologous sequences have been implemented to predict RNA secondary structures [7-9]. However, these approaches all depend on well-aligned RNA sequences and thus are very vulnerable to alignment errors.

In this paper, we design a seed-based method for prediction of RNA secondary structure using unaligned sequences. Our method is motivated by BLAST [10] and completes the prediction with the aid of Mfold [4] [11-12]. Given a single RNA sequence or multiple unaligned sequences, we first detect all possible stems in each sequence using the position matrix. The matrix is composed of 0 or 1 and indicates the paired or unpaired information for each position in the sequence. Then, we select some seeds for RNA folding by finding and assessing possibly conserved stems across all sequences. Specially, we assess the stems using HMM for single sequence and *Profile*-HMM [13] for multiple sequences. Finally, we use Mfold to perform the final prediction on the basis of selected seeds. We tested our method on the data sets of known ncRNA sequences downloaded from the Rfam database [14]. The comparison with Mfold shows that our method can predict RNA secondary structure with a higher accuracy than Mfold.

2 Methods

Our method works with stems and predicts secondary structure in three major steps:

1) Detecting all possible stems in an RNA sequence.
2) Selecting the seeds by finding and assessing possible conserved stems.
3) RNA secondary structure prediction based on the seeds.

2.1 Detecting All Possible Stems in an RNA Sequence

The idea for detecting the stem is motivated by ddbRNA [15], which is a faster algorithm for detecting conserved secondary structures in both pairwise and multiple DNA sequence alignments. However, ddbRNA does not take GU and UG pairs into account and not allow gaps in the alignments. Furthermore, ddbRNA possibly make more mistakes due to the parameter of K and the approach for shuffling alignments. Here, we improve it by taking GU and UG pairs into account and by assessing the detected stems. We first introduce the so-called position matrix which can be constructed from the RNA sequence. Then we describe how to get the potential stems by scanning the matrix row by row.

Given an RNA sequence of length N, Seq, we build one $N \times N$ position matrix (denoted by M^{Seq}) containing 0 or 1. This can be done by following steps:

1) The reverse complement of *Seq*, *Seq'*, is firstly obtained from the original sequence. Specially, the following rules should be obeyed:
   - The complement of 'G' is not 'C' but the set of {C, U}. For simplicity, we denote φ = {C, U}.
   - The complement of 'U' is not 'A' but the set of {A, G}. For simplicity, we denote ψ = {A, G}.
   - To get the reverse complement of *Seq*, we scan it from the end of the sequence to the beginning.
2) We build one $N \times N$ matrix containing *Seq'* in the first row. The ith ($0 \le i \le N$-1) row contains the sequence generated from *Seq'* by shifting i position to the left (circular left shift).
3) The position matrix M^{Seq} is built by comparing *Seq* with each row of the matrix mentioned above. For the ith row of M^{Seq}, 0 or 1 is assigned to the jth ($0 \le j \le N$-1) position by comparing the jth base of *Seq* with the jth base of the ith row in the matrix. Here, '1' means the corresponding position is paired and '0' means the position is unpaired. The following two rules should be obeyed when two bases are compared with each other:
   - If b equals b', then 1 is assigned.
   - If b does not equal to b', then 0 is assigned.

Here, b is a base from *Seq* and b' is an element from the matrix. Specially, when b' is φ or ψ, the word "equal" means that b belongs to b'. As shown in Figure 1, One $N \times N$ matrix (the left) is firstly constructed from the original sequence. Then the position matrix, M^{Seq}, is built by comparing *Seq* with each row of the left matrix.

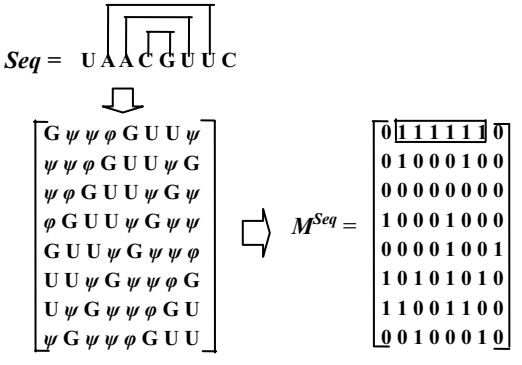


Fig. 1. The construction of the position matrix for an RNA sequence

We can detect all possible stems in an RNA sequence by scanning its position matrix row by row. The key is to find all zones of continual non-zero in the matrix. There is a one-to-one mapping between the stems and the zones of continual non-zero in all rows of the matrix. As shown in Figure 1, one stem in the RNA sequence (*Seq*) are mapped to one zone of continual non-zero in its matrix (M^{Seq}). Finally, we point that the time complexity of the approach mentioned above is O (N^2).

2.2 Selecting the Seeds for RNA Folding

Finding Conserved Stems Across Multiple Unaligned Sequences
To find conserved stems across multiple sequences, we compare all the stems in the sequences according to their matrices. This can be done by dealing with two stems each time. The comparing of two stems is performed by multiplying their matrices. Here, the matrix for a stem means the position matrix for the sequence corresponding to the stem. This matrix also can be computed by the method shown in Figure1. Therefore, the position matrix for two aligned stems can be built by multiplying their matrices. Note that the *i*, *j* element of the resulting matrix is computed by:

$$M\,[i, j] = M_1\,[i, j] \times M_2\,[i, j] \tag{1}$$

Here, M_1, M_2 are the original matrices and M is the resulting matrix. For simplicity, we still describe the multiplying of the matrices with following equation:

$$M = M_1 \times M_2 \tag{2}$$

Note that the dimensions of all original matrices in equation (2) should be identical. It is easy to be disposed when the length of the original sequences for two stems is the same one, because their matrices have the same dimension in this case. In other cases, we only dispose the so-called compatible stems which share a certain number of paired positions at the core. In this case, we just add some gaps (denoted by '_') into the shorter sequence at the beginning and the end. Then we construct a new matrix for the new sequence by following revised method.

To revise the method shown in Figure 1, we add the following rule to the step 1) in Section 2.1:

- The complement of a gap is also a gap.

And we add the following rule to the step 3) in Section 2.1:

- If *b* or *b'* is '_', or both of them are '_', then 0 is assigned.

As shown in Figure 2, *Stem1* and *Stem2* are easily aligned by directly multiplying M^{Stem1} and M^{Stem2}. As for *Stem1* and *Stem3*, we first verify they are compatible because they share two paired positions at the core. Then we add two gaps into *Stem3* and construct a new matrix for it. Finally, we align *Stem1* and *Stem3* by multiplying M^{Stem1} and M^{Stem3}. Suppose D is an alignment of n stems and M^{Stemi} is the matrix of the ith stem, then the matrix for D (denoted by M^D) can be computed by following equation:

$$M^D = \prod_{i=1}^{n} M^{Stemi} \tag{3}$$

Obviously, the multiplying of the matrices satisfies the commutative law and the associative law. To find possible conserved stems, we also search for zones of continual non-zero in the resulting matrix as we do in section 2.1. As shown in Figure 2, the conserved stem across *Stem1* and *Stem2* can be indicated by the two rectangles in matrix $M^{Stem1} \times M^{Stem2}$. The conserved stem across *Stem1* and *Stem3* can be indicated by the rectangle in the matrix $M^{Stem1} \times M^{Stem3}$. We point that the time complexity of the

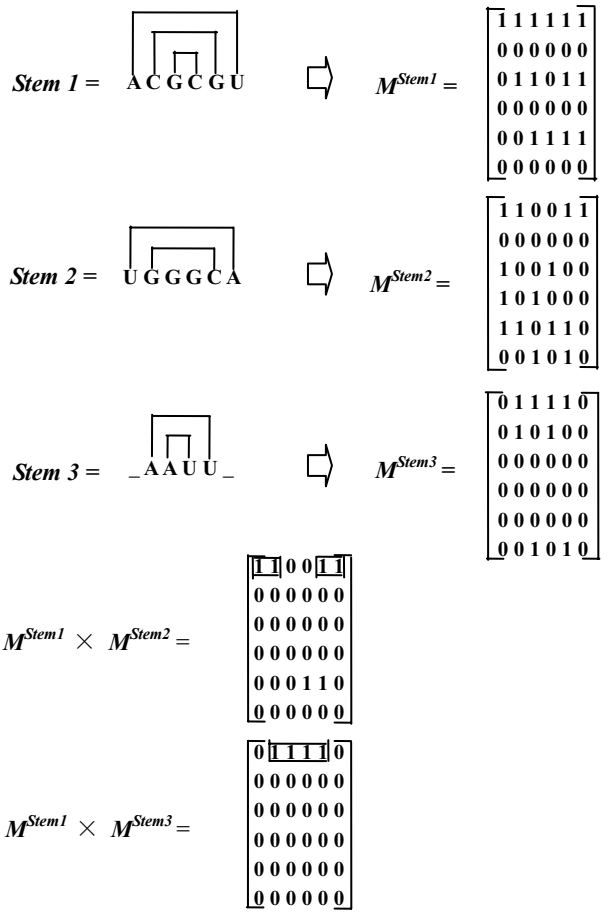


Fig. 2. The alignment of stems for finding conserved stems across multiple sequences

equation (2) is still O (N^2), where N is the length of the stem. Therefore, the time complexity of the equation (3) is O ((n-1) N^2).

Actually, to detect conserved stems across n sequences, we first select n stems from these n sequences (one stem for one sequence) and then apply the equation (3). We repeat this process until all stems from all sequences are chosen and disposed. Suppose m stems are selected from each sequence on average and the average length of the stem is l. Therefore, the time complexity of this process is approximately O (m^n (n-1) l^2). In practical application, to reduce the time cost, we keep m be a little constant.

Selecting the Seeds by Assessing Conserved Stems

We use the *Signal-to-Noise* to assess the conserved stems. The major steps for assessing a conserved stem are described as follows:

1) For anyone of the conserved stems detected by the approach mentioned above. We record its length (i.e. the number of pairs) and the sequence alignment of the conserved stem (i.e. the sequence fragments of the aligned stems).
2) We generate a randomized alignment from the original sequence alignment of the conserved stem.
3) We extract all sequences from the newly generated alignment and detect possible conserved stems across them using the method mentioned before.
4) Let the length of the original conserved stem be the *Signal* and the length of the possible conserved stem in the randomized alignment be the *Noise*. The *Signal-to-Noise* (i.e. the ratio of the *Signal* to the *Noise*) is thus computed for them. Note that, we set *Noise* to be 1 if there are no conserved stems in the randomized alignment, and we set *Noise* to be the maximum if there are more than one conserved stems.

We select some conserved stems with higher *Signal-to-Noise* as the seeds for RNA folding. We compute the *Signal-to-Noise* for each conserved stem across all original sequences using the approach mentioned above. Then, we sort them in decreasing order of *Signal-to-Noise* and select the stems with *Signal-to-Noise* greater than the given threshold as the seeds for RNA folding.

The most difficulty in the approach mentioned above is how to get a randomized alignment from the original alignment. We accomplish this purpose by randomly permuting the columns of the original alignment. We control the permutation process using a probabilistic method. Given an RNA alignment, we compute the probability of it using the *forward* algorithm of *Profile*-HMM [13].

Suppose the producing model for an alignment of n sequences (denoted by 'D') is given by a five-tuple of model components $\lambda = \{S, V, \pi, T, E\}$, where λ is the *Profile*-HMM, S is the set of states, V is the set of observed characters, π is the initial probability vector of the states, T is the matrix of *transition* probabilities, and E is the matrix of the *emission* probabilities. We define them as follows:

- $S = \{Cod, ncRNA, Oth\}$, where *Cod* refers to the protein-coding region, *ncRNA* refers to the non-coding region, and *Oth* refers to other regions.
- $V = \{\text{A, C, G, U, \_}\}^n$.
- $\pi = \{\pi_{Cod}, \pi_{ncRNA}, \pi_{Oth}\}$, where π_{Cod} is the initial probability of *Cod*, π_{ncRNA} is the initial probability of *ncRNA*, and π_{Oth} is the initial probability of *Oth*.
- $T = [t\ (w, w')]$, where w and w' belong to S and $t\ (w, w')$ is the *transition* probability from w to w'. Obviously, T is a 3×3 matrix.
- $E = [e_w(x)]$, where w belongs to S, x belongs to V, and $e_w(x)$ is the *emission* probability of producing x on state of w. Obviously, E is a 3×5^n matrix.

We set α as the recursion variable and initialize it with following equation:

$$\alpha_0(w) = \pi_w e_w(o_0) \qquad w \in S,\ o_0 \in V \tag{4}$$

Here, o_j $(0 \le j \le N\text{-}1)$ denotes the jth column of D, and N is the length of D.

The recurrence process is simplified with following equation:

$$\alpha_j(w') = \sum_{w \in S} \alpha_{j-1}(w)\, t(w, w')\, e_{w'}(o_j) \tag{5}$$

$$w, w' \in S,\ o_j \in V,\ 1 \le j \le N\text{-}1$$

The probability of D, P ($D|\lambda$), is computed with following equation:

$$P(D|\lambda) = \sum_{w \in S} \alpha_{N-1}(w) \tag{6}$$

We repeat the permutation until the common difference between the probability of the new alignment and that of the original alignment is less than a certain threshold (for example 0.001). In this way, we get a randomized alignment from the original alignment.

It should be pointed that the approach mentioned above is only suitable for multiple unaligned sequences. When the input is a single RNA sequence, we assess the detected stems based on HMM and also select several of them as the seeds for RNA folding. The approach for this case is similar to the approach mentioned above except that V should be changed to {A, C, G, U}.

2.3 Structure Prediction Based on the Seeds

We predict RNA secondary structures on the basis of seeds and the Mfold program. We first find some good seeds using the approach described in Section 2.2 and then align all sequences according to the seeds. Then we divide the alignment into many segments such that one seed is at the beginning of one segment. After that, we predict secondary structures for each segment and finally assemble them into an integral secondary structure. As for the prediction of each segment, we use the seed in it as the starting position for folding and complete the final prediction using Mfold program. In more detail, we respectively use Mfold to fold each sequence of the segment and finally select the structure with maximum base-pairs as the result. As shown in Figure 3, the three-way alignment is divided two segments and one segment includes one seed.

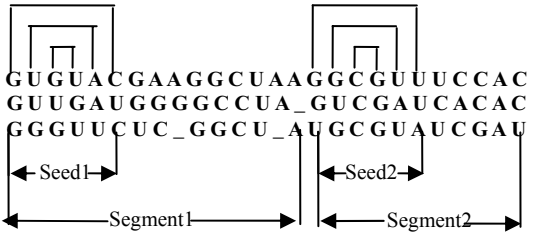


Fig. 3. RNA secondary structure prediction based on seeds. Mfold is used to fold each section on the basis of corresponding seeds.

3 Results

3.1 Test Data Sets

We constructed one single-sequence data set, one two-sequence data set and one three-sequence data set based on Rfam7.0. To build the single-sequence data set, we downloaded 80 RNA sequences from Rfam. This data set is also used as the input for Mfold. To build the two-sequence data set and the three-sequence data set, we first downloaded 80 pairwise alignments and 80 three-way alignments from Rfam. Then we extracted two sequences from each pairwise alignment and three sequences from each three-way alignment. Also, we randomly extract one sequence from each downloaded alignment and use it as the input for Mfold. Specially, the downloaded sequences belong to multiple different RNA families annotated by Rfam.

To test the performance of out method and compare it with Mfold, we also downloaded the secondary structure from Rfam for each alignment and compute the *accuracy* (measured as sensitivity) and *error* (measured as false positive rate) of predicting base-pairs. We compute the *accuracy* as the number of true positives divided by the sum of true positives + false negatives, the *error* as the number of false positives divided by the sum of true positives + false positives. Note that the sum of *accuracy* + *error* is not necessarily 1.

3.2 Tests for RNA Sequences with Known Secondary Structures

The results for single-sequence data set are compared with Mfold, using the parameters suggested by the authors of [11-12] and are reported in Table 1. The results for multiple RNA sequences are also compared with Mfold and are reported in Table 2 and Table 3. Note that the second and third columns in Table 1 through Table 3 are the results for our method.

As shown in Table 1, our method has comparable *accuracy* and *error* with Mfold for length ≤150bp, while has higher *accuracy* and lower *error* than Mfold for length ›150bp. Our method has average *accuracy* 69.93% and average *error* 17.55% for single-sequence test as a whole. As shown in Table 2 and Table 3, our method has average *accuracy* 72.97% and average *error* 19.56% for two-sequence test, average *accuracy* 79.27% and average *error* 25.03% for three-sequence test. Generally, our method exhibits marked higher *accuracy* and comparable *error* with Mfold in the case of multiple unaligned sequences.

Table 1. *Accuracy* and *error* on single-sequence data set

Length (bp)	*accuracy* (%)	*error* (%)	*accuracy*.Mfold (%)	*error*.Mfold (%)
≤100	69.02	18.02	69.65	18.16
100-150	70.89	16.16	71.01	16.68
150-200	71.16	15.21	69.56	20.16
≥200	68.65	20.82	65.16	26.78
total	69.93	17.55	68.85	20.45

Table 2. *Accuracy* and *error* on two-sequence data set (Id, percentage identity)

Id (%)	*accuracy* (%)	*error* (%)	*accuracy*.Mfold (%)	*error*.Mfold (%)
40-60	72.02	20.21	68.16	22.11
60-70	75.87	21.34	69.52	25.18
70-80	72.08	18.65	68.35	19.28
80-100	71.92	18.02	70.98	19.64
total	72.97	19.56	69.25	21.55

Table 3. *Accuracy* and *error* on three-sequence data set (Id, percentage identity)

Id (%)	*accuracy* (%)	*error* (%)	*accuracy*.Mfold (%)	*error*.Mfold (%)
40-60	78.15	25.01	70.98	24.11
60-70	79.76	24.63	70.65	24.38
70-80	79.18	25.81	69.56	26.02
80-100	79.99	24.65	70.05	24.16
total	79.27	25.03	70.31	24.67

4 Discussion

We design, implement and evaluate a seed-based method for predicting common RNA secondary structures in unaligned sequences. Our method detects conserved stems using the position matrix, assesses conserved stems using the *Profile*-HMM, selects the seeds using the *Signal-to-Noise*, and uses the seeds as the constraint for Mfold. Our method is based on Mfold and improves it by the seed-extending idea. Furthermore, the fact that our method takes unaligned sequences as input makes it immune from alignment errors and thus be more suitable for practical application. Finally, the algorithms for detecting and assessing conserved stems also make our method differ from other methods.

Despite the limited amount of data, we have shown in the experiments that our method can predict RNA secondary structures with a better performance. In the case of the single-sequence test, our method has a higher accuracy than Mfold, especially for longer RNA sequences. This should be the correct behavior, since we divide a longer sequence into several sections according to the seeds and respectively fold each section using Mfold. In the case of the multiple-sequence tests, our method exhibits a marked higher accuracy than Mfold, especially when more RNA sequences are used. This is true because we detect and assess the conserved stems using multiple homologous sequences and then select some of them as the seeds for folding. As a result, we improve Mfold by using these seeds as the constraint for RNA folding.

There are several ways in which our method could be improved. One potential improvement could be assessing the stem using a combination of MFE and probabilistic models. To do this, we can respectively evaluate the energy and the *Signal-to-Noise* of the stem, and then assign each of them a weight. Thus, the stem can be assessed by both MFE and probabilistic methods. This possibly improves the

accuracy of selecting the seeds and thus betters the accuracy of structure prediction. Another way to improve our method might be changing the algorithm for generating randomized alignments. In this paper, we directly permute the columns of the alignment to get the randomized alignment. Actually, this approach might not take the randomicity of the original sequence into account and thus possibly gets bad results. Another alterative approach for this could be firstly permuting the columns of each sequence and then aligning the new sequences using standard multiple sequence alignment tools such as BLAST or CLUSTALW [16]. Finally, the time complexity of our method is probably high, especially for the process of finding conserved stems. Therefore, to speed up the method, we can detect, compare and assess the stems in a parallel way. But this needs further research to maintain the accuracy of the method.

Acknowledgments. This work has been supported by the National Natural Science Foundation of China under Grant No. 60673018. The authors wish to thank the anonymous referees for their helpful comments.

References

1. Eddy, S.R.: Non-coding RNA genes and modern RNA world. Nat RevGenet 2(12), 919–929 (2001)
2. Huttenhofer, A., Schattner, P., Polacek, N.: Non-coding RNAs:hope or hype? TRENDS in Genetics 21(5), 289–297 (2005)
3. Furtig, B., et al.: NMR spectroscopy of RNA. Chembiochem 4(10), 936–962 (2003)
4. Zuker, M., Stiegler, P.: Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information. Nucleic Acids Research 9(1), 133–148 (1981)
5. Hofacker, I., et al.: Fast folding and comparison of RNA secondary structures. Monatshefte fur Chemie 125(2), 167–188 (1994)
6. Gardner, P.P., Giegerich, R.: A comprehensive comparison of comparative RNA structure prediction approaches. BMC Bioinformatics 5, 140–157 (2004)
7. Hofacker, I., Fekete, M., Stadler, P.: Secondary structure prediction for aligned RNA sequences. Journal of Molecular Biology 319(5), 1059–1066 (2002)
8. Ruan, J., Stormo, G., Zhang, W.: An iterated loop matching approach to the prediction of RNA secondary structures with pseudoknots. Bioinformatics 20(1), 58–66 (2004)
9. Knudsen, B., Hein, J.: Pfold: RNA secondary structure prediction using stochastic context-free grammars. Nucleic Acids Research 31(13), 3423–3428 (2003)
10. Altschul, S., et al.: Basic local alignment search tool. Journal of Molecular Biology 215(3), 403–410 (1990)
11. Zuker, M.: Mfold web server for nucleic acid folding and hybridization prediction. Nucleic Acids Research 31(13), 3406–3415 (2003)
12. Mathews, D., et al.: Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure. Journal of Molecular Biology 288(5), 911–940 (1999)
13. Durbin, R., et al.: Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. Cambridge University press, Cambridge (1998)

14. Sam, G.J., Simon, M., Mhairi, M.: Rfam: annotating non-coding RNAs in complete genomes. Nucleic Acids Research 33(suppl. 1), D121–D124 (2005)
15. Bernardo, D.d., Down, T., Hubbard, T.: ddbRNA: detection of conserved secondary structures in multiple alignments. Bioinformatics 19(13), 1606–1611 (2003)
16. Thompson, J., Higgins, D., Gibson, T.: CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, positions-specific gap penalties and weight matrix choice. Nucleic Acids Research 22(22), 4673–4680 (1994)

Automatic Segmentation of Neoplastic Hepatic Disease Symptoms in CT Images

Marcin Ciecholewski and Marek R. Ogiela

AGH University of Science and Technology, Institute of Automatics
30 Mickiewicza Ave., 30-059 Krakow, Poland
{ciechol,mogiela}@agh.edu.pl

Abstract. In this paper will be described a new method of automatic segmentation of inflammation and neoplastic hepatic disease symptoms, visible in computed-tomography (CT) images. The liver structure will be at first extracted from the image using the approximate contour model. Then, the appropriate histogram-based transformations will be proposed to enhance neoplastic focal changes in CT images. For segmentation stage of cancerous symptoms, the analyzed images will be processed using binary morphological filtration with the application of a parameterized mean defining the distribution of pixel gray-levels in the image. Then, the edges of neoplastic lesions situated inside the liver contour are localized. To assess the efficiency of the proposed processing procedures, experiments have been carried out for two types of tumours: haemangiomas and hepatomas. The experiments were conducted on 60 cases of various patients. In this set 30 images showed single and multiple focal hepatic neoplastic lesions, and the remaining 30 images show the healthy organ. Experimental results confirmed that the proposed method is an efficient tool which may be used in the diagnostic support procedures for normal and abnormal liver. The efficiency of proposed algorithm reach the level of over 83% of correct recognition of pathological changes.

1 Introduction

Liver disease is one of the most widespread type of lesions in the world [3]. For example the average incidence of liver cancer is 16 cases per 100,000 population worldwide, and 4 cases in Europe [3]. In most cases liver cancer manifests itself in abnormal cells whose growth leads to the emergence of either single or multiple neoplastic lesion formations. If the hepatic tumor is detected early, the progress of the disease can be quickly stopped. Consequently, the design and development of Computer-Aided Diagnosis systems and techniques can help doctors improve their diagnosis. Computer support early detection of pathological changes play an important role in screening and specialist examinations. The Artificial Intelligence and Computational Intelligence algorithms have recently been playing a huge role in the development of new classes of intelligent systems supporting medical diagnoses . Some of them can even be used to interpret the meaning of medical images [10]. However, in many cases, apart from the search for the semantic and medical conclusions of the analysed images, the systems must also conduct the preliminary processing and

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 414–421, 2007.

analysis of the images in question to detect pathological lesions present in the image. In CT images, these lesions may be single or multiple and their correct detection forms an important step which determines the subsequent ability to identify the tumour type and its progression [1, 5, 8].

In this paper will be proposed a new method which facilitates the automatic segmentation of single and multiple neoplastic lesions of various shapes and locations within the liver structure. Research was conducted on 60 different cases obtained during medical CT examinations for various patients.

2 Cancerous Symptoms Segmentation in CT Liver Images

In this section will be presented the method for automatic segmentation of single and multiple neoplastic lesions visible in liver CT images. In this procedures at first, the liver contours is extracted from the image. Next , the histogram is transformed to enhance, and improve visibility of neoplastic lesions in CT images. For the proper segmentation of neoplastic changes, the images are processed using binary morphological filtration operators with the application of a parametrized mean defining the distribution of gray levels of pixels in the image. The last step is to locate the edges of neoplastic lesions found inside the liver contour.

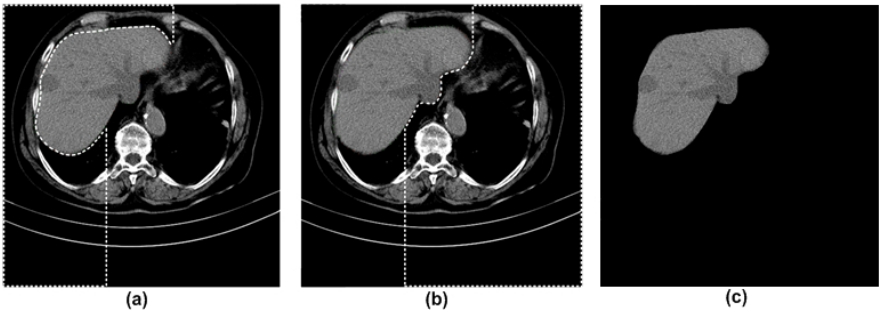


Fig. 1. Liver contour detection in a CT image of the abdominal cavity. (a) A CT image with a superimposed polygon containing the approximate liver edge and a fragment of the left side of the image. (b) A CT image with a superimposed polygon containing the approximate liver edge and a fragment of the right side of the image. (c) The segmented liver structure.

The first step in the sequence of neoplastic lesion detection is to distinguish the liver structure, as this allows surplus elements to be removed from the image. The liver structure is segmented from the CT image by first finding the liver contour made up of a finite number of joint polylines which approximate particular fragments of the liver edge in the computed tomography image. This approach is presented in detail in paper [2]. The area of the image located outside the liver contour is divided into two polygons which are eliminated from the image. Fig. 1.c shows an example of a CT image with the liver structure segmented using the approximate contour method [2].

Having segmented the liver structure in the CT image, the next step leading to the extraction of neoplastic foci is the transformation of the histogram by its equalisation, which yields an even distribution of the number of pixels relative to the gray levels. The histogram is a single-dimensional statistical function obtained by counting the number of pixels corresponding to specific grey levels. Let g: $M^2 \rightarrow Z$ be the gray-level CT image containing the liver structure and $(x,y) \in [0,M-1] \times [0,M-1]$ be the pixel coordinates. Then: $g(x,y) \in Z$. The histogram, $h(k)$: $Z \rightarrow Z$ is defined as the following set:

$$h(k)=\{(x,y): g(x,y)=k\} . \quad (1)$$

where k is the value of the grey level. Let $g_{HE_q}: M^2 \rightarrow Z$ be the histogram equalisation transformation. It is represented by the following relationship:

$$\forall (x,y) \in M^2 \; g_{HE_q}(x,y)=LUT(g(x,y)),\; LUT(k)=(g_{max}/(x_{max}*y_{max}))*\Sigma h(j) \; (j=0,,k) . \quad (2)$$

where *LUT*(look-up table) is the adjustment table allowing the gray levels of the input image to be changed in accordance with the values stored in the table. The g_{max} variable is the number of grey levels in the image, while the $x_{\max} * y_{max}$ product defines the size of the image frame ($x_{\max} * y_{max}=M^2$), and $h(j)$ is the number of pixels with the j gray-level.

Let g_{GF}: $M^2 \rightarrow Z$ be the Gaussian smoothing function [6] with the mask G with the dimensions of $5x5$, $g_{GF}=(g \times G)(x,y)$. As a result of the subsequent transformations g_{HE_q} and g_{GF}, we obtain an image in which neoplastic foci will be detected. The image is defined by the following formula:

$$g_{GF} = (g_{HE_q} \times G)(x,y) . \quad (3)$$

The details of these operations are shown in Fig. 3, which presents the subsequent histogram transformations. Figures 2 (a) and 2 (b) contain the image of the liver structure g and its histogram $h(k)$.

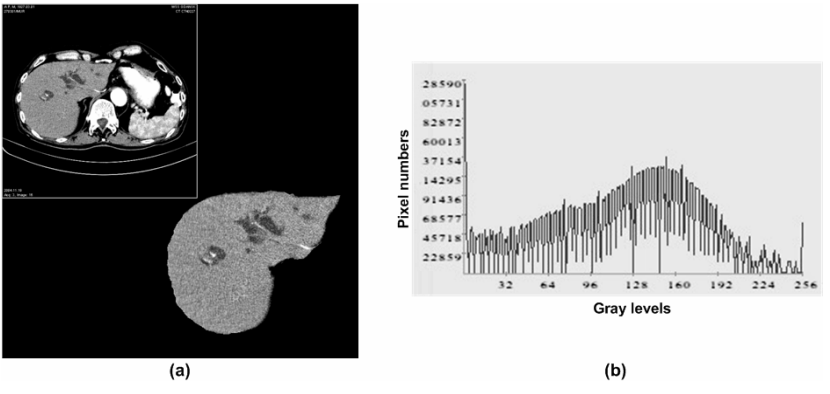


Fig. 2. CT image of the liver. (a) An image containing the CT scan of liver and separated structure (b) A histogram $h(k)$.

Figures 3 (a) show the g_{HE_q} image after the histogram equalisation transformation and the equalised histogram graph. Figures 3 (b) contain the g_{GF} image after Gaussian smoothing and the equalised histogram graph.

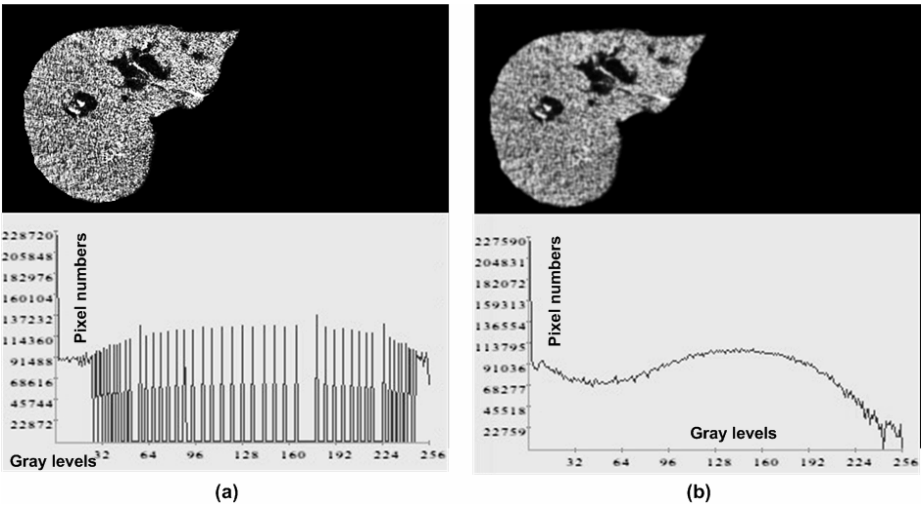


Fig. 3. Transformations in the CT image of the liver. (a) The image and histogram after equalisation transformation. (b) The image and histogram after Gaussian smoothing.

To segment neoplastic lesions, the image g_{GF} is first binarised using the function b: $M^2 \rightarrow M^2$ where

$$b=\{255 \text{ (white) for } g_{GF} > \mu, \textit{ and } 0 \text{ (black) for } g_{GF} \leq \mu\}. \tag{4}$$

The $\mu=\Sigma\, k*(h(k)/M^2$ (for $k \in Z$) parameter defines the mean calculated for the pixels of the g_{GF} image. Then, the binary image b is subjected to elementary filtration operations such as erosion, median filtration and edge detection.

Let the e: $M^2 \rightarrow M^2$ function define erosion and $e^i(b)$: $M^2 \rightarrow M^2$ be the iterative erosion repeated i times (*ie*). Under the assumption that S*E* is the structural element with the dimensions of 3*x*3, we can also define the median filtration operation. Let m: $M^2 \rightarrow M^2$ be the median filtration with the S*E* mask. Also, let b_L: $M^2 \rightarrow M^2$ define the linear filter using a Laplacian to detect the edges in the image. The Laplacian is represented by the L mask with the dimension of 3*x*3. Every filtration is defined using image algebra operators for binary images [4, 9].

The definitions are as follows:

Erosion and iterative erosion

$$E=\{b\text{-}SE\},\ e^i(b)=\{\ldots((b\text{-}SE)\text{-}SE)\text{-}SE\}. \tag{5}$$

$$\text{Median } m=\{b \circ SE\}. \tag{6}$$

$$\text{Laplacian } b_L=\{b \times L\}. \tag{7}$$

The research has shown that the best method to improve neoplastic lesion segmentation is double erosion $e^2(b)$. As a result, the black image areas are magnified. Then, the image edges $e^2(b)$ are smoothed using median filtration. As a result of applying that filtration, we get an image defined by the following formula: $m=\{e^2(b) \circ SE\}$. Figure 4 (a) shows a binarised specimen liver image, while Figure 4 (b) shows a liver image after double erosion and median filtration.

Fig. 4. The binary morphological filtering (a) The binary image (b) The image after double erosion and median filtration

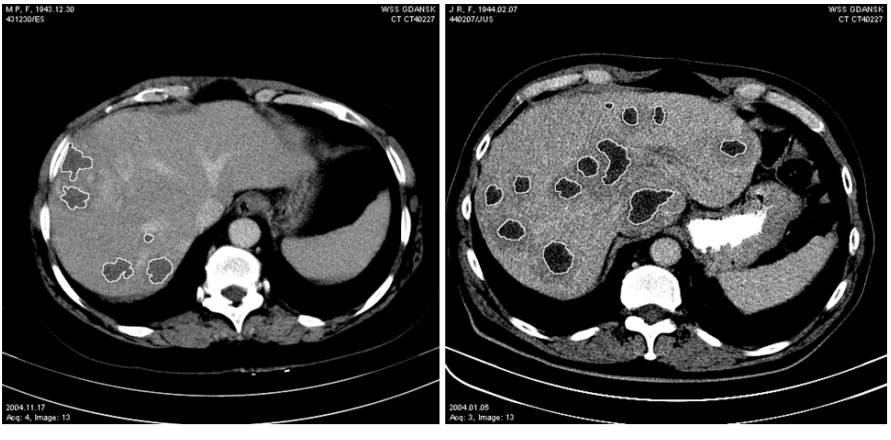


Fig. 5. The example of neoplastic lesion segmentation in hepatic CT images

As a result of using the linear filter $b_L=\{m \times L\}$, we get a set comprising a sub-set representing a specific number of edges of neoplastic lesions $b_f=\{f_1,f_2,...,f_K\}$ and also the liver contour c. Thus $b_L=\{f_1,f_2,...,f_K,c\}$. The liver contour can be eliminated, since its coordinates have been calculated during the liver structure segmentation, as presented in paper [2]. The set of neoplastic lesion edges b_f is superimposed on the original CT image of the abdominal cavity and the liver, Fig. 5. Thus is the neoplastic lesion segmentation achieved. Figure 5 shows the example liver images with the segmented neoplastic lesions.

3 Results

For the analysis of CT images, the material from the Department of Image Diagnostics of the Provincial Specialist Hospital in Gdansk, Poland, was used. The CT images were obtained with Somatom Emotion 6 (Siemens) scaner. To verify the suitability of the proposed algorithms, tests were run for two types of tumours: haemangiomas and hepatomas, and for images showing healthy livers. The tests were conducted on 60 cases of various patients. Half of them contained single or multiple neoplastic lesions of various shapes and locations within the liver structure. Another thirty showed healthy livers. For every CT image, the neoplastic lesion segmentation was classified to one of the four basic classes cases:

- TN - true negative,
- FP - false positive,
- FN - false negative and
- TP - true positive [6].

Table 1 shows the results of classifying the processed images in relation to the number of categories.

Three ratios: sensitivity, specificity and accuracy, were calculated for the test data obtained. The sensitivity represents the number of patients who have neoplastic lesions: $Sensitivity=TP/(TP+FN)$. Specificity defines the number of patients with no neoplastic lesions: $Specificity=TN/(TN+FP)$. Accuracy is defined as: $Accuracy=(TP+TN)/(TP+TN+FP+FN)$.

Table 1. Test results for 60 CT images of the abdominal cavity and the liver

Lesion	Quantity	TN-true negative	FP-false positive	FN-false negative	TP-true positive
Hemangioma	15	0	0	3	12
Hepatoma	15	0	0	5	10
Normal Liver	30	25	5	0	0
Total Number	60	25	5	8	22

The sensitivity of the method for test data placed at the level 0.733, its specificity to 0.833, and accuracy to 0.783. The ratios obtained confirm that the method is highly suitable for image diagnostics of the liver. However, it should be noted that the value of the FT parameter was obtained for images in which the Falciform ligament which splits the liver into the right and left lobes was highly visible, which means that it could be interpreted as one of the possible foci. In the case of multiple neoplastic lesions whose foci had a diameter below 1 cm, FN values occurred for only some of them.

4 Conclusion

In the article was presented a new method of automatic segmentation of single and multiple neoplastic lesions in CT images for a computer system assisting the early

diagnostics of inflammation and neoplastic lesions of the liver. The segmentation of these lesions was achieved by first segmenting the liver from the CT image. Then, the histogram was transformed to enhance the neoplastic lesions in CT images. In order to segment the neoplastic lesions, the images were processed using binary-morphological filtration operators with the application of a parametrised mean defining the distribution of grey levels of pixels in the image. The next step was to locate the edges of neoplastic lesions found inside the liver contour. To verify the suitability of the proposed method, tests were conducted for two types of tumours: haemangiomas and hepatomas, and also for images of healthy livers. The test data yielded the following ratio values: sensitivity 0.733, specificity 0.833 and accuracy 0.783. In the future, research work will aim at developing a method which eliminates the Falciform ligament as a lesion, because the current method classifies this ligament as a false positive (FP). Another research objective will be to raise the sensitivity of the method to reduce the occurrence of FNs and thus to improve the results of the segmentation of multiple neoplastic lesions below 1 cm in diameter.

All the foci diagnoses obtained will also be used as feature vectors in procedures for interpreting the images researched run by a system which is currently being developed to automatically analyse medical images. The analysis will facilitate determining not only the number and location of lesions detected, but will also be aimed at drawing medical conclusions about their occurrence and will help in formulating the diagnosis using tests conducted with other modalities as well as results of functional tests.

Acknowledgements

This research project was financed with state budget funds for science in 20062007 as research project no. 3T11F 030 30 of the Ministry of Science and Higher Education.

References

1. Chen, E.L., Chung, P.C., Chen, C.L., Tsai, H.M., Chang, C.I.: An automatic diagnostic system for CT liver image classification. IEEE Transactions on Biomedical Engineering 45(6), 783–794 (1998)
2. Ciecholewski, M., Debski, K.: Automatic Segmentation of the Liver in CT Images Using a Model of Approximate Contour. In: Levi, A., Savaş, E., Yenigün, H., Balcısoy, S., Saygın, Y. (eds.) ISCIS 2006. LNCS, vol. 4263, pp. 75–84. Springer, Heidelberg (2006)
3. Danaei, G., Vander Hoorn, S., Lopez, A.D., Murray, C.J., Ezzati, M.: Causes of cancer in the world: comparative risk assessment of nine behavioural and environmental risk factors. Lancet 366, 1793–1874(2005)
4. Meyer-Base, A.: Pattern Recognition for medical imaging. Elsevier Academic Press, Amsterdam (2004)
5. Park, S., Seo, K., Park, J.: Automatic Hepatic Tumor Segmentation Using Statistical Optimal Threshold. In: Sunderam, V.S., van Albada, G.D., Sloot, P.M.A., Dongarra, J.J. (eds.) ICCS 2005. LNCS, vol. 3514, pp. 934–940. Springer, Heidelberg (2005)

6. Rangayan, R.M.: Biomedical signal analysis. Wiley Computer Publishing, New York (1997)
7. Ritter, G.X., Wilson, J.N.: Computer Vision Algorithms in Image Algebra. CRC Press, Boca Raton (2000)
8. Seo, K.: Automatic Hepatic Tumor Segmentation Using Composite Hypotheses. In: Kamel, M., Campilho, A. (eds.) ICIAR 2005. LNCS, vol. 3656, pp. 922–929. Springer, Heidelberg (2005)
9. Sonka, M., Fitzpatrick, J.M.: Handbook of Medical Imaging. In: Medical Image Processing and Analysis, vol. 2. SPIE Press, Bellingham, WA (2000)
10. Tadeusiewicz, R., Ogiela, M.R.: Medical Image Understanding Technology. Springer, Heidelberg (2004)

A Robust Localization Method for Mobile Robots Based on Ceiling Landmarks

Viet Thang Nguyen[1], Moon Seok Jeong[1], Sung Mahn Ahn[2], Seung Bin Moon[1], and Sung Wook Baik[1,*]

[1] College of Electronics and Information Engineering, Sejong University, Seoul 143-747, Korea
{n_vietthang,msjeong,sbmoon,sbaik}@sejong.ac.kr
[2] College of Business Administration, Kookmin University, Seoul 136-702, Korea
sahn@kookmin.ac.kr

Abstract. In this paper, we propose a robust method for the process of localization of a mobile robot through a vision system. The mobile robot is a compact system consisting of an embedded board and a fish-eye camera. The fish-eye camera looks upward to capture ceiling images. The camera provides a sequence of images for the process of detection and tracking ceiling features. These features are used like natural landmarks to detect the state of translation and rotation of the robot. Our method requires less computational power and resources for a robot, and thus can be used at home. The results produced in this study showed the advantages of our method in terms of both speed and accuracy.

1 Introduction

In recent years, intelligent mobile robots have been applied to a variety of fields. These robots are able to perform tasks traditionally done by human in industry, military, entertainment and home assistance. These include toy robots, guide robots, surveillance robots and household robots all of which are able to self-navigate in indoor and outdoor environments without human manipulation. A mobile robot must have the ability to accurately localize itself in a modified environment and build a map by itself for further manipulation. Most existing methods are based on laser range finders, sonar range finders and vision systems. The laser range finder detects obstacles and measures distances with high frequency data. However, laser sensors use an energy beam that can be dangerous. Therefore, adequate precautions must be taken to avoid damage to the eyes. The sonar range finder is useful for mobile robots to navigate and recognize their environment. This method allows robots to measure distance to objects. This method can be easily performed but requires a silent environment for manipulation without noise from other sources. The vision system obtains data on a given room or space. Then, the robot uses the data to navigate or

* Corresponding author.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 422–430, 2007.

localize in the given area. In some cases, the robots are able to interact with people through the vision-based human interaction.

One of the most popular robots is the intelligent household mobile robot which works in an indoor environment. There are two main vision approaches applied to the mobile robot. Those are ceiling vision [1], [2], [8] and forward vision [4], [5], [6], [7], [8]. Forward Vision focuses on detecting objects located in front of the robot to find direction in a modified environment. Ceiling Vision finds the robot's location in an indoor environment. Therefore, the robot is able to know where it is and how to accurately calculate the best path to its destination. For household automatic mobile robots, previous research has sought to setup a landmark system and use it for self-localization in an indoor environment [1], [2]. Others attempted to detect objects in front of the robot to navigate in modified environment [3], [4], [5], [6], [7].

In this paper, we introduce a new method of self-localization by constructing a natural landmark system based on the invariance of ceiling features on sequence images. The mobile robot is composed of a compact mainboard and a fish-eye camera. For environmental configuration, the mobile robot runs in the room while the fish-eye camera captures ceiling images and sends them to the robot's processor. The process of localization generates grouping feature points from a sequence of ceiling images. A feature point is referred to as a corner detected from input images. Each group of feature points satisfying the threshold is defined as a ceiling landmark.

This study is organized as follows. Section 2 discusses previous research. The localization algorithm is explained in section 3. Section 4 presents the results of our research. Section 5 draws conclusions from our research and discusses potential research related to this paper for the future.

2 Related Work

Many researchers and engineers have worked on the landmark systems for intelligent mobile robots. To achieve this goal, feature detection is required beforehand. Several methods and solutions have been proposed for this work [9], [10], [11], [12], [13], [14]. However, these methods are expensive and require a great deal of resources.

Some researchers have proposed autonomous environment mapping, localization and navigation for indoor mobile robots using monocular vision and multiple 2D pattern tracking [1], [2]. In this way, the camera is vertically oriented toward the ceiling and not necessarily calibrated. The environment map is a mosaic on a 2D pattern that is detected on the ceiling plane using natural landmarks. This algorithm enables the mobile robots to reproduce trajectories defined by key images representing virtual memory. The pattern tracker is a contour model that takes into account image gray scale level variations. It uses the condensation algorithm to efficiently track the 2D pattern on a cluttered background.

Some researchers have presented the Monte Carlo Localization based on the condensation algorithm [6], [7]. In this method, a visual map of the ceiling is obtained by the use of mosaicing and the localization of the robot using simple scalar brightness measurement as sensor input.

Se et al. have researched vision-based mobile robot localization and a mapping system. SIFT (Scale Invariant Feature Transform) obtains visual landmarks in

unmodified environments [4], [5]. By keeping SIFT landmarks in a database, the robot tracks the landmarks over time and builds a 3D map of its environment, while also using these 3D landmarks for localization.

Jeong et al. [8] presented a method of setting a landmark system using Ceiling Vision - based on the Simultaneous Localization and Mapping techniques. This method collects the effective corners from ceiling images captured by a mobile robot using the Harris algorithm. Then, the Kalman filter algorithm is applied to these corners to create a 3D landmark system. This method has high fidelity but requires heavy computation and processing. Thus, the method is not suitable for a mobile robot equipped with low-cost components and computing power.

To overcome the aforementioned limitations, this paper introduces a new method of localization for mobile robots to build ceiling landmark systems. This method uses a sequence of ceiling images for the fast detection of feature points. This makes it easier to identify feature points in each image by analyzing the robot's rotation and translation.

3 Ceiling Vision-Based Localization

In this section, we explain the basic idea behind our vision-based approach for the localization of a mobile robot that operates in an indoor environment. The purpose of this is to extract ceiling features from the sequence images and detect the state of translation and rotation. At a translation state, the features on the ceiling are detected and used as natural landmarks. We use the Susan corner detection method [9] for detecting feature points. This method is one of the fastest algorithms in feature point detection, since it does not require complex calculation. Because the float could not be supported on the embedded board that we tested [Fig. 4], we chose the Susan method. The Susan method in feature point detection is not as good as others such as the Harris algorithm, but it detects points faster. With a cheap camera which may become soiled after long-term use, the quality of captured images will be not perfect. For these reasons, we present a method to optimize the calculation of translation and rotation detection without complex calculation. The robot has 2 types of movement (rotation and translation) that are not in a curved line as in Fig. 1. Thus, we separated the movements of the robot into 2 states (rotation and translation) needed for detection.

- Translation: In translation action, the robot always moves forward or backward following a line, not a curve. Therefore, with the same object feature captured in two sequence images, there is some change on the y axis of the image coordinates but no change on the x axis.
- Rotation: When the robot wants to change direction, the robot will rotate on an angle and then continue moving. Therefore, with the same object feature captured in a sequence of two images, they are always the same distance from the image center with only a different angle.

We can accomplish curve action by repeatedly combining the two types of movement mentioned above.

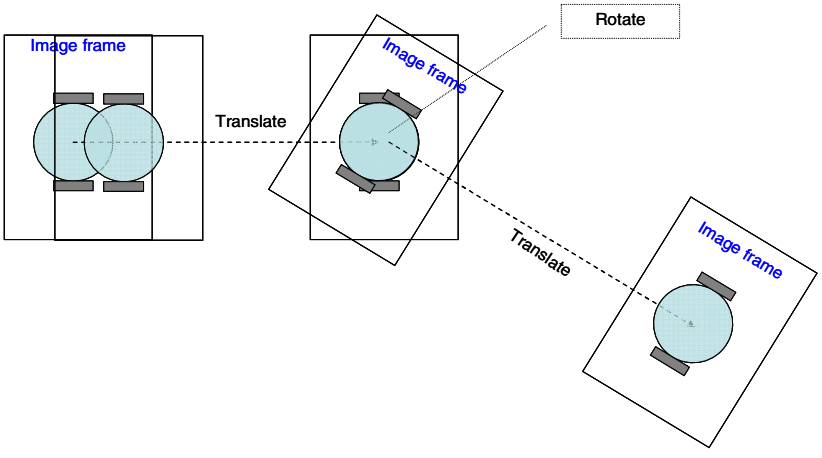


Fig. 1. Robot movement analyzing

3.1 Rotation Detection

As shown in Fig. 2, with features detected from two sequence images, we detected the rotation angle of the mobile robot. The process can be described as follows:

1. We found and matched features in the two images that have the same vector length. Then, we calculated the angles between the matched features.
2. We calculated the mean value of the rotation angles. The mean angle 0 indicates no rotation on the two images.

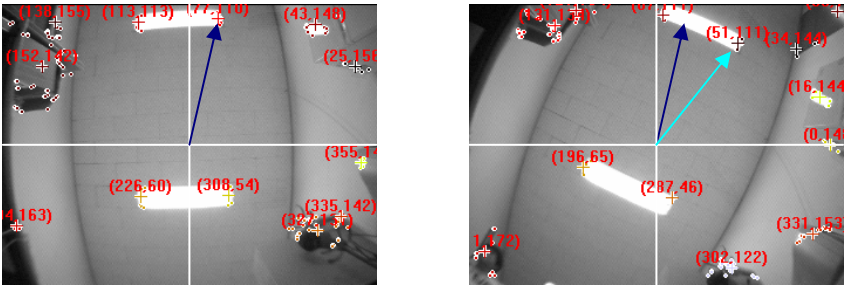


Fig. 2. Feature Point detection and vectorial calculation in 2 rotation image frames

Fig. 2 shows two vectors (the dark arrow and the light arrow). They are from image centers of the same ceiling features. Their lengths are the same (110 in the first and 111 in the second) but they have different angles (77 in the first and 51 in the second). The rotation angle is (26 = 77 - 51).

3.2 Translation Detection

We analyzed the translation in multi frame images shown in Fig. 2 After the rotation detection step, we calibrated the features. Some algorithms calibrate the image before feature detection. This will create an image with a lot of noise which result in the poor performance of feature detection. We detected features in actual images and calibrated the feature points. This results in better performance and lessens processing time. The process is as follows:

1. After feature points have been calibrated, we can match certain groups clustering features that appeared in the first image with ones in the second image. According to the results, one will notice a disparity between two different positions of the same feature group. The disparity provides us with information for translation. There is some change on the y axis of the image coordinates but no change on the x axis. If the change is less than a quarter of the image size on y, there is no problem in translation.

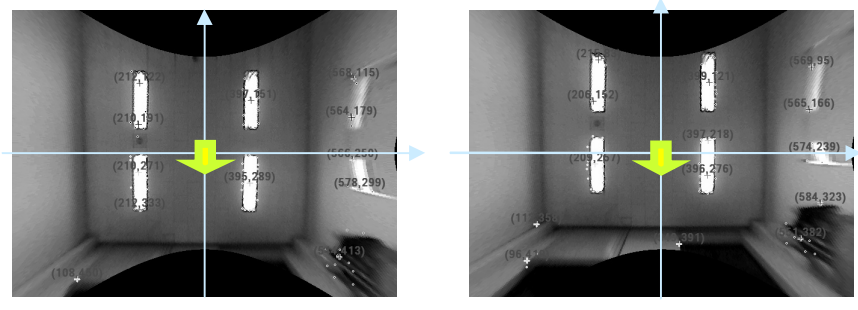


Fig. 3. Feature Point detection and grouping in 2 translated image frames

Fig. 3 shows the grouping result using Susan corner detection in two sequence images. The first pairs of numbers in parenthesis are on the x position and the second on the y position of the group center in the original image.

4 Experimental Results

The proposed method was tested with the Harris, Susan and Fast corner detection algorithms. We used an embedded board (Fig. 4), which has the following specifications: PXA270 CPU(32bit Xscale) 624 MHz, 32 MB SDRAM memory, 64 MB Flash memory and a gray fish eye camera with 130^{o} view angle and 320x240 resolution for each image viewed after lens calibration. We tested this method with several ceiling images captured from several different indoor environments such as a laboratory, an apartment, and a lobby. This was carried out in two steps. The first was for the point feature detection process including the Harris, Susan, and Fast Corner detection algorithms. Then, we compared the results as well as the elapsed time of each process in order to choose a suitable algorithm for our requirements. In the second step, we used the method proposed in this study to detect rotation and translation movement from sample sequence images in zigzag path tests.

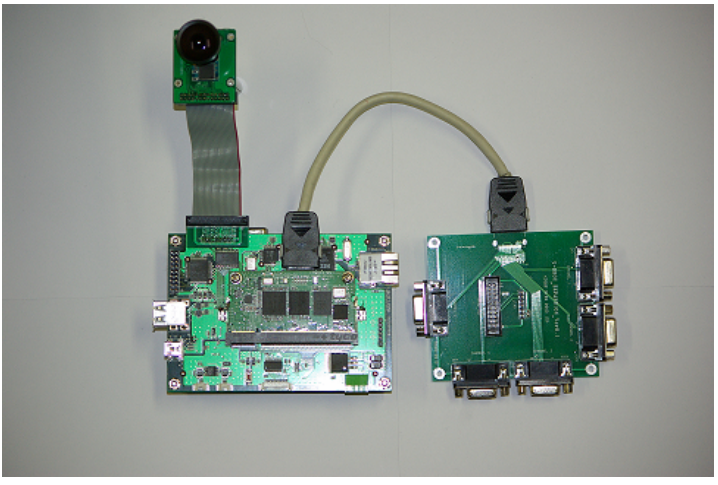


Fig. 4. The PXA270 embedded board

Fig. 5 is the ceiling map of the test room. We tested zigzag paths consisting of randomized rotation angles and translated distances.

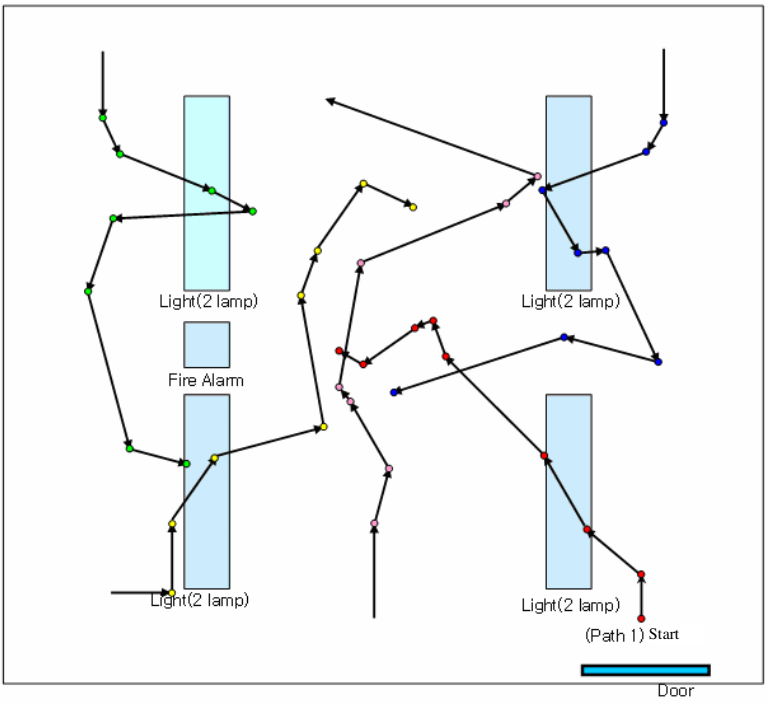


Fig. 5. The Experimental test room plan

Table 1 shows the translation and rotation calculated for each of the two sequence images using the Susan method to detect feature points. The table shows the distance between the two camera positions in pixel units, the angle of the rotation state, and the elapsed time for the detection process of each zigzag path that was tested. Table 2 shows errors in translation and rotation detection between calculated results and the real states of zigzag paths. It includes the moving distance average/max error (cm), the rotation angle average/max error (degree) and the real movement (moving distance and rotated angle) in max errors.

Table 1. This table represents the results from the detection process

	Path 1		Path 2		Path 3		Path 4		Path 5	
	Actual distance	Calculation distance	Actual distance	Calculation distance	Actual distance	Calculation distance	Actual distance	Calculation distance	Actual distance	Calculation distance
1	30cm	31.5 cm	40cm	43.75cm	50cm	52.5cm	45cm	45.5cm	63cm	64.75cm
2	-50°	-47°	-90°	-89°	30°	29°	-25°	-25°	15°	15°
3	50cm	50.75cm	46cm	45.5cm	22cm	22.75cm	27cm	28cm	38cm	38.5cm
4	20°	20°	35°	41°	40°	38°	-43°	-45°	-45°	-46°
5	56cm	54.25cm	50cm	52.5cm	73cm	70cm	66cm	68.25cm	51cm	50.75cm
6	-15°	-13°	40°	40°	-100°	-85°	5°	5°	-10°	-10°
7	92cm	94.5cm	75cm	75.25cm	48cm	49cm	30cm	28cm	10cm	10.5cm
8	25°	24°	-85°	-82°	-65°	-63°	150°	147°	50°	48°
9	24cm	22.75cm	88cm	91cm	18cm	17.5cm	93cm	94.5cm	83cm	82.25cm
10	80°	79°	30°	28°	70°	67°	-68°	-65°	58°	57°
11	14cm	14cm	30cm	31.5cm	82cm	64.75cm	52cm	49cm	105cm	98cm
12	-25°	-24°	15°	15°	130°	129°	-34°	-35°	-18°	-18°
13	46cm	47.25cm	55cm	50.75cm	64cm	64.75cm	109cm	101.5cm	27cm	28cm
14	65°	64°	80°	66°	-33°	-34°	-60°	-59°	-120°	-110°
15	17cm	17.5cm	37cm	40.25cm	120cm	113.75Cm	40cm	42cm	150cm	131.25cm
Elapsed time		15.2sec		15.1sec		15.2sec		15.2sec		15.1sec

Table 2. This table shows errors in the detection process between the calculated results and real states of tests of zigzag paths

	Moving Distance Avg. error	Rotation angle Avg. error	Moving Distance Max. error	Rotation angle Max. error	Real movement in max errors	
					Moving distance	Rotation angle
Room 1	2.77cm	2.4°	18.75cm	15°	150cm	100°
Room 2	2.84cm	2.4°	19.25cm	14°	125cm	120°
Room 3	2.78cm	2.3°	17.5cm	12°	168cm	110°
Room 4	2.77cm	2.5°	19.25cm	16°	142cm	95°
Room 5	2.77cm	2.4°	18.75cm	15°	150cm	100°

5 Conclusion and Future Work

The method proposed in this work was tested with a low cost embedded board running on the Linux embedded system and a gray fish eye camera to capture 320x240 images. We captured many sequence images and run processes to detect the change of the position of the camera in direction and angle rotation. We also showed how to detect which features are not on the ceiling as well as how to use ceiling features to build 2D/3D land masks.

In future research, we plan to integrate this method with an intelligent control robot. With intelligent control, the robot will build and update land masks automatically when transferred to other indoor environments.

References

1. Blanc, G., Aït Aïder, O., Mezouar, Y., Chateau, T., Martinet, P.: Indoor Navigation of Mobile Robot: An Image Based Approach. In: Proceedings of the 35th International Symposium on Robotics, ISR'04, Paris, France, March 23-26th, 2004, p. 010 (2004)
2. Aït Aïder, O., Chateau, T., Lapresté, J.T.: Indoor autonomous navigation using visual memory and pattern tracking. In: British Machine Vision Conference, Kingston, England, September 2004, vol. 2, pp. 657–665 (2004)
3. Brady, M., Wang, H.: Vision for mobile robots. Phil. Trans. R. Soc. Lond. B 337, 341–350 (1992)
4. Se, S., Lowe, D.G., Little, J.J.: Vision-Based Global Localization and Mapping for Mobile Robots. IEEE Transaction on Robotics, vol. 21(3) (June 2005)
5. Se, S., Lowe, D.G., Little, J.J.: Mobile Robot Localization and Mapping with Uncertainty using Scale-Invariant Visual Landmarks. The International Journal of Robotics Research 21(8), 735–758 (2002)
6. Thrun, S., Bennewitz, M., Burgard, W., Cremers, A.B., Dellaert, F., Hähnel, D.F.D., Rosenbeg, C., Roy, N., Schulte, J., Schulz, D.: MINERVA: A Second – Generation Museum Tour- Guide Robot. In: Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference, May 10-15, 1999, vol. 3, pp. 1999–2005 (1999)
7. Dellaert, F., Burgard, W., Thrun, S.: Using the CONDENSATION Algorithm for Robust, Vision-based Mobile Robot Localization. In: Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 23-25, 1999, vol. 2, Pages: Digital Object Identifier 10.1109/CVPR.1999.784976 (1999)
8. Jeong, W.Y., Lee, K.M.: CV-SLAM: A new Ceiling Vision-based SLAM technique. In: IEEE/RSJ International Conference on Intelligent Robots and Systems 2005, pp. 3195–3200 (2005)
9. Smith, S.M., Brady, J.M.: SUSAN - a new approach to low level image processing. Int. Journal of Computer Vision 23(1), 45–78(34) (1997)
10. Harris, C., Stephens, M.: A combined edge and corner detector. In: Proceedings of the 4th Alvey Vision Conference 1998, vol. 15, pp. 147–151(1998)
11. Tomasi, C., Kanade, T.: Detection and Tracking of Point Features. Tech Report CMU-CS-91-132 Carnegie-Mellon University, Pittsburgh, PA (1991)

12. Shi, J., Tomasi, C.: Good Features to Track. In: 9th IEEE Conference on Computer Vision and Pattern Recognition, June 1994, pp. 593–600. IEEE Computer Society Press, Los Alamitos (1994)
13. Trajkovic, M., Hedley, M.: Fast corner detection. Image and Vision Computing 16(2), 75–87 (1998)
14. Zitová, B., Flusser, J., Kautsky, J., Peters, G.: Feature point detection in multiframe images. In: Svoboda, T. (ed.) Czech Pattern Recognition Workshop, Peršlák, Czech Republic, February 2-4, 2000, pp. 117–122 (2000)

Robust Parameter Decision-Making Based on Multidisciplinary Knowledge Model

Jie Hu[1], Yinghong Peng[1], and Guangleng Xiong[2]

[1] School of Mechanical Engineering, Shanghai Jiao Tong University,
200240 Shanghai, P.R. China
{hujie,yhpeng}@sjtu.edu.cn
[2] Department of Automation, Tsinghua University,
100080 Beijing, P.R. China
glxiong@cims.tsinghua.edu.cn

Abstract. This paper presents an approach of multidisciplinary knowledge modeling and robust parameter decision-making. Firstly, a multidisciplinary knowledge model is established. Secondly, a multidisciplinary decision-making algorithm is presented to solve the model. Finally, the method is demonstrated by a design example, in which knowledge in mechanics, cybernetics and dynamics are addressed. The results prove that multidisciplinary knowledge modeling is feasible, and the proposed method can be applied in multidisciplinary parameter decision-making process.

Keywords: Parameter decision-making, Knowledge model, Multidisciplinary design.

1 Introduction

The development process of a complex product may be involved with "time dimension" concurrence and "space dimension" collaboration. The overall performance of a complex product generally depends on a number of specifications distributed in multi-teams from different disciplines. Many researchers paid attention to multidisciplinary collaborative design approaches. Ahn and Kwon [1] presented an efficient reliability-based multidisciplinary design optimization (RBMDO) strategy. Rosenman *et al.* [2] put forward a 3D virtual world environment, which provides real-time multi-user collaboration for designers in different locations and allows for the different design disciplines to model their view of a building as different representations. Lee and Jeong [3] described the development of a decomposition based design (DBD) method using optimal sensitivity information with respect to coupled or interdisciplinary design variables. Chen and Jin [4] analyzed characteristics of multidisciplinary collaborative design (MCD) of product and proposed a new MCD-oriented product information model (MCDPM) that integrates physical structure, design semantic and collaboration management data. Kong *et al.* [5] developed an Internet-based collaboration system for a press-die design process for automobile

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 431–442, 2007.

manufacturers with CORBA, Java, Java3D and a relational database system. Yin *et al*. [6] presented an approach to component-based distributed cooperative design over the Internet where an extended multi-tier model (Browser/Server) is used to implement the web-based remote design system. Wang *et al*. [7] developed of a distributed multidisciplinary design optimization (MDO) environment (called WebBlow) using a number of enabling technologies including software agents, Internet/Web, and XML. Gantois and Morris [8] described a quite innovative multidisciplinary optimisation method based on robust design techniques. Giassi *et al*. [9] described a quite innovative multidisciplinary optimisation method based on robust design techniques: MORDACE (multidisciplinary optimisation and robust design approaches applied to concurrent engineering). Bai *et al*. [10] introduced the concept of the PLF (Product Layout Feature) and provided a solution to the problems of PLF modeling. As a result of the solution, collaborative design activities among multi-teams from different disciplines can be consistently carried out on PLF models in the PDM environment.

The problem with these researchers is that they focused on the coordination and optimization in concurrent and collaborative design. However, the most important problem for concurrent and collaborative design is that how to obtain the design knowledge from multidisciplinary domain and make decision for multidisciplinary parameters.

A multidisciplinary knowledge modeling method to support robust parameter decision-making is introduced in this paper. It collects the design knowledge from multidisciplinary specification to construct a formulated model for concurrent and collaborative design. Then, an approach of robust parameter decision-making is presented and is illustrated by a bogie design example.

2 Multidisciplinary Knowledge Model

The overall performance of a complex product generally depends on a number of specifications distributed in various disciplines. Design knowledge for parameter decision-making can be expressed as a set of constraints, which can be divided into two classes: specification constraints and relation constraints. The former is design goals, including the requirements and limitations on product performance, shape size and so forth, which are determined by user requirements before starting design process. The latter is relationships between specifications and design variables which can be obtained from design principles in each discipline. In traditional design, the information in specification constraints are not made full use, only treated as evaluated criteria.

Specification constraints are design goals, which are almost defined early during the design process, so the successful ratio for passing simulation can be improved with determining the design variable solution space by considering all the prouct specifications simultaneously, and selecting values in this consistency region.

Therefore, the fundamental task of parameter design is to determine solution space and detect potential conflicts to ensure all the constraints are satisfied, and then obtain the optimized parameter specifications.

2.1 Uncertainties in Multidisciplinary Parameter Decision-Making

In multidisciplinary parameter decision-making, two kinds of imprecise information can be defined: 1) stochastic noise and 2) parameter uncertainty. Stochastic noise such as machining errors can be described with probability functions. However, the parameter uncertainty, which varies as the design cycle proceeds, cannot easily be described by the statistical robust theory. These parameters can only be estimated within known bounds in the early design phase and become fixed in the final stage. If the process is described with traditional robust theory, the probability function will become thinner and thinner as the design proceeds. However, detailed information in each interval such as the probability function is difficult to obtain in concurrent and collaborative design.

For parameter uncertainty, it has long been recognized as a topic worthy of investigation within system theory and artificial intelligence. In this paper, the rough regions from which design variables could choose values are described by interval boxes, which are effective to use filtering algorithm to detect inconsistent conflicts with interval description manner. Moreover, interval box is the important part of the following robust parameter decision-making model.

2.2 Multidisciplinary Parameter Decision-Making Model

Before introducing the multidisciplinary parameter decision-making model, several important concepts will be defined.

Uncertain Parameter: Some parameters cannot be known precisely early in the design period, but have known bounds. The uncertainties in these parameters decreases as the design proceeds and the bounds of the parameter will converge to a fixed value as the design is finalized.

Design Variable: Design parameters (variables) are those variables which can be determined independently by the designers.

Objective Variable: The design goal. Usually objective variables are definite before product development begins, but they may change for a complex product with a long development time.

Solution: A solution is a certain design variable vector satisfying all of the constraints for the given specifications.

Solution Space: A solution space consists of a set of design variable intervals. However, the solution space may not contain a solution.

Parameter Decision-making Model: In multidisciplinary parameter design, constraints with explicit mathematical forms are used to verify and analyze the parameter uncertainties during the design process.

Robust: With parameter uncertainties in multidisciplinary parameter decision-making model, if a design structure described by the model can satisfy the given specifications, the design structure is robust within the given bounds of the uncertain parameters.

The multidisciplinary parameter decision-making model is formulated as:

$$z(t)\in\left[z(t)^{\mathrm{L}},z(t)^{\mathrm{U}}\right],\boldsymbol{x}\in\left[\boldsymbol{x}^{\mathrm{L}},\boldsymbol{x}^{\mathrm{U}}\right]$$
$$\text{s.t.}\begin{cases}\boldsymbol{g}(z,\boldsymbol{x})=0 \quad \boldsymbol{g}=\left[g_1,g_2,\cdots,g_p\right]^{\mathrm{T}}\\ \boldsymbol{h}(z,\boldsymbol{x})\le 0 \quad \boldsymbol{h}=\left[h_1,h_2,\cdots,h_q\right]^{\mathrm{T}}\\ \dfrac{dz}{dt}=\boldsymbol{f}(z,\boldsymbol{x},t),\ \boldsymbol{f}=\left[f_1,f_2,\cdots,f_n\right]^{\mathrm{T}}\\ z(0)\in\left[z_0^{\mathrm{L0}},z_0^{\mathrm{U0}}\right]\\ \boldsymbol{x}\in\left[\boldsymbol{x}^{\mathrm{L0}},\boldsymbol{x}^{\mathrm{U0}}\right]\end{cases} \tag{1}$$

where $z(t)\in\boldsymbol{R}^n$ is a state variable vector of an n-dimensional continuous dynamical system, $\boldsymbol{x}\in\boldsymbol{R}^m$ is an m-dimensional design variable vector; $\boldsymbol{g}$ and $\boldsymbol{h}$ are constraint vectors of equations and inequalities respectively, and $\boldsymbol{f}$ is an nth-order ordinary differential equations (ODEs) vector describing the behavior of a dynamical system. $\boldsymbol{g}$, $\boldsymbol{h}$ and $\boldsymbol{f}$ compose the set of constraints. $[z_0^{\mathrm{L0}},z_0^{\mathrm{U0}}]$ and $[\boldsymbol{x}^{\mathrm{L0}},\boldsymbol{x}^{\mathrm{U0}}]$ are the initial intervals. $[z(t)^{\mathrm{L}},z(t)^{\mathrm{U}}]$ and $[\boldsymbol{x}^{\mathrm{L}},\boldsymbol{x}^{\mathrm{U}}]$ are consistent intervals filtered by the following robust parameter decision-making algorithm. If the domain of any variable in $\boldsymbol{z}$ or $\boldsymbol{x}$ vector is empty, it means that there exists conflicts in current design project and some specifications cannot meet the requirements.

3 Robust Parameter Decision-Making Algorithm

Some important concepts and notations are introduced first before discuss the robust parameter decision-making algorithm framework.

1) *G*-interval and interval box

An *G*-interval written as $\boldsymbol{G}=\boldsymbol{R}\cup\{-\infty,+\infty\}$, is an interval extended by real number set $\boldsymbol{R}$. $\forall a,b\in\boldsymbol{G}$, the interval [*a*, *b*] means $\{x\in\boldsymbol{G}\,|\,a\le x\le b\}$. $\boldsymbol{I}(\boldsymbol{G})$ represents the set that contains all real intervals on the *G*-interval.

An interval box $\boldsymbol{B}$ is defined as a Cartesian product of *n* *G*-intervals:

$$\boldsymbol{B}=\boldsymbol{I}_1\times\boldsymbol{I}_2\times\cdots\times\boldsymbol{I}_n,\quad \boldsymbol{I}_i\in\boldsymbol{I}(\boldsymbol{G})\quad i=1,...,n \tag{2}$$

2) Interval extension

Let $g:\boldsymbol{R}^n\to\boldsymbol{R}$ be a mapping, if when

a) $\boldsymbol{I}_1, \boldsymbol{I}_2, \cdots, \boldsymbol{I}_n \in \boldsymbol{I}(\boldsymbol{G})$ and
b) $\forall r_1 \in \boldsymbol{I}_1, r_2 \in \boldsymbol{I}_2, ..., r_n \in \boldsymbol{I}_n$ are satisfied, there exists a relationship:

$$g(r_1, r_2, ..., r_n) \in \boldsymbol{G}(\boldsymbol{I}_1, \boldsymbol{I}_2, ..., \boldsymbol{I}_n) \tag{3}$$

then $\boldsymbol{G} : \boldsymbol{I}(\boldsymbol{G})^n \to \boldsymbol{I}(\boldsymbol{G})$ is defined as the interval extension of the mapping g, which is not formally unique according to the definition.

Two important steps are employed in the algorithm framework: approximation and narrowing, with which variable intervals in the constraints network could converge into steady interval boxes. These interval boxes contain solution space as a subset and are consistent with the given specifications.

3) Approximation

For every relation ρ on the real number set $\boldsymbol{R}$, the approximation of ρ is denoted as $apx(\rho)$, which is the smallest G-interval that contains the relation ρ.

4) Narrowing

Let ρ be a relation on the real number set $\boldsymbol{R}$, the narrowing function of ρ is denoted as $\vec{\rho} : \boldsymbol{I}(\boldsymbol{G})^n \to \boldsymbol{I}(\boldsymbol{G})$, for every interval box $\boldsymbol{u}$,

$$\vec{\rho}(\boldsymbol{u}) = apx(\rho \cap \boldsymbol{u}) \tag{4}$$

where relation ρ can be either an interval box or a mapping. When ρ is a mapping, $\rho \cap \boldsymbol{u}$ is the intersection with $\boldsymbol{u}$ and the result of mapping ρ being carried out on $\boldsymbol{u}$.

For the approximation and narrowing functions, there exist some properties characterizing their behaviors on G-interval:

For interval boxes $\boldsymbol{u}$, $\boldsymbol{v}$, and a relation ρ, if $\boldsymbol{u}, \boldsymbol{v}, \rho \in \boldsymbol{I}(\boldsymbol{G})^n$, such properties are satisfied:

$$\vec{\rho}(\boldsymbol{u}) \subset \boldsymbol{u} \tag{5}$$

$$\rho \cap \boldsymbol{u} = \vec{\rho}(\boldsymbol{u}) \cap \rho \tag{6}$$

$$\boldsymbol{u} \subset \boldsymbol{v} \Rightarrow \vec{\rho}(\boldsymbol{u}) \subset \vec{\rho}(\boldsymbol{v}) \tag{7}$$

$$\vec{\rho}(\vec{\rho}(\boldsymbol{u})) = \vec{\rho}(\boldsymbol{u}) \tag{8}$$

Equation (5) shows the convergence of a narrowing function. After narrowing function is carried out, the narrowed interval box will be smaller than the initial interval. Equation (6) implies the correctness that no valid design solutions will be discarded. If some conflicts are detected in the design process, it is certain that no solutions exist in the current design space. Equation (7) can assure the monotonicity of narrowing functions. Equation (8) represents the uniqueness with which the approximation and narrowing functions are necessary to be executed once in the current state, which avoids the infinite computing iterations.

In this paper, an algorithm combined with genetic algorithm is put forward to resolve the design model. In order to use the algorithm to solve the design model, the formation of constraints must be uniformed to equalities, so the inequalities and ordinary differential equations (ODEs) should be transformed.

1) Transform of the inequalities

Construct a temporary vector $\boldsymbol{\theta}$ for inequality constraint vector $\boldsymbol{h}$, $\boldsymbol{\theta} \in \boldsymbol{R}^q$ and $\boldsymbol{\theta} \geq 0$, with the help of, inequalities can be transformed into equalities.

$$\boldsymbol{h}(z, \boldsymbol{X}) \leq 0 \Rightarrow \boldsymbol{h}(z, \boldsymbol{X}) + \boldsymbol{\theta} = 0 \tag{9}$$

2) Transform of ordinary differential equations

Considering the ordinary differential equations (ODE) vector $\boldsymbol{f}$ in Eqs. (1),

$$\begin{cases} \dot{z} = \boldsymbol{f}(z, \boldsymbol{x}, t) \quad z(t) \in R^n \quad \boldsymbol{x} \in R^m \\ z(0) = z_0 \in \left[z_0^{\mathrm{L0}}, z_0^{\mathrm{U0}} \right] \end{cases} \tag{10}$$

If $\boldsymbol{f}$ is not dependent on time t which is coincident with engineering design requirement generally, Eqs. (10) could be rewritten as

$$\dot{z} = \boldsymbol{f}(z, \boldsymbol{x}) \tag{11}$$

whose solution is formulated as

$$z(\boldsymbol{x}, t, z_0) = \left[z_1(\boldsymbol{x}, t, z_0), ..., z_m(\boldsymbol{x}, t, z_0) \right]^{\mathrm{T}} \tag{12}$$

Eq. (12) shows the relationship between one of the system state variables and the its associated design variables. But in engineering design, it is difficult to obtain the analytical solution of Eq. (12), therefore some numerical methods are adopted to deal with Eq. (12) and obtain a series of values of state variables at some discrete time points:

$$z_{n+1} = z_n + \boldsymbol{f}'(z_n, \boldsymbol{x}, h) \tag{13}$$

where h is the iteration step size. Actually Euler method, Runge-Kutta method and so on can be used to evaluate the derivatives of $\boldsymbol{f}$ at a fixed time point t^*, then Eqs. (10) can be transformed to a group of algebra equations as result.

The robust parameter decision-making algorithm used to solve the model is described as:

```
Procedure ModelResolve( In {C=[g,h,f],x0,z0}; Out {x, z})
Begin
  Transform inequality set h into equality set g1
  Transform ODE set f into equality set g2
  For t*=initial_time To final_time
    = Call IntervalResolve({x0, z0},{g,g1,g2})
    IF([x,z]=NULL) Then exit //conflict and exit
    EndIf
```

```
    x → x_0
    For each state_variables z_i
      z_i_max(t*)=max_GA(x, t*, z_0)
      z_i_min(t*)=min_GA(x, t*, z_0)
    EndFor
    Keep x, z_max(t*), z_min(t*) at the time point of t*
    t*= t*+h    // h is the step length of iteration
  EndFor
End
Procedure IntervalResolve(x, C(x))
Begin
  Boolean changed // a flag to judge the changes of
                     variable intervals
    While changed = TRUE Do changed← FALSE
      For j=1 TO p+q+n Do //aiming at all the
                           constraints
        For i=1 To q+m+n Do //aiming at all the
                             variables in a constraint
          Approximate and narrow the constraints C_j(j=
          r,...,t,1≤r≤t≤p+q+n) which x_i should satisfy
          Approximate and narrow the interval consistent
          with the domain of (x_1,...,x_{i-1},x_{i+1},...,x_{q+m+n})
          If (the domain of x_i is empty)
            Then show conflict messages, Return NULL
          EndIf
          If (the interval of x_i is changed)
            Then changed = TRUE
              Replace the old interval of xi with the
              newly obtained interval
          EndIf
        EndFor
      EndFor
    EndWhile
  Return x
End
```

The procedure of ModelResolve() is the main procedure to solve the model. The procedure of IntervalResolve() is used to solve the consistent space of design variables at every discrete time point t^*. In this algorithm, it implements two functions max_GA() and min_GA() with genetic algorithm to compute the maximum and minimum of state variables. Using genetic algorithm will improve the correctness of approximation and narrowing functions with sacrificing complexity of the algorithm.

The aim of the above algorithm is to obtain the consistent space of design variables. Then, designers can use optimization method to obtain optimized solution in the consistent space.

In the parameter decision-making algorithm, a conflict means that the given box has no solution, which means that the current design cannot satisfy all of the network constraints. The constraints network should then be revised until the consistency checking is satisfied. With these three conclusions, the algorithm can be used to check for conflicts, which provides valuable support to concurrent and collaborative design.

According the arithmetic, conflict can be predicted by the state of the consistency interval. If consistent space of a design variable is empty set, then there is conflict in the constraint network, and conflict information is presented.

4 Example

Following is an example of bogie, which is shown in Fig. 2, to illustrate how to design parameter based on multidisciplinary knowledge model. The bogie of a railway car is mainly composed of suspension, wheels and axle boxes, in which the design of dumping system including suspension spring and buffer spring is a key problem and involved with multidiscipline, such as mechanics, cybernetics and dynamics.

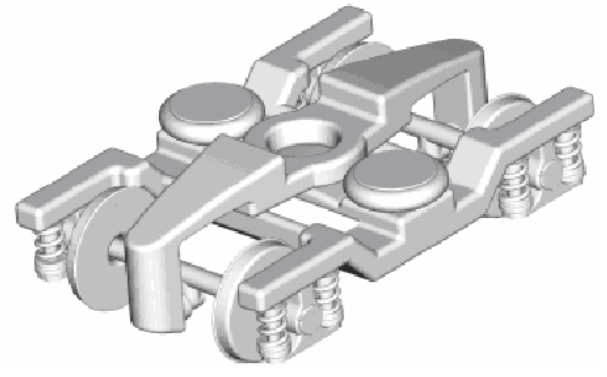


Fig. 2. An example of the railway vehicle bogie

The knowledge in bogie dumping system design is illustrated as follows.
Step 1. Multidisciplinary knowledge model.
1) Knowledge in mechanics can be expressed as

$$[\tau] - 8P_{\max} D_n C_{mn} / \pi d_n^{\,3} - temp_1 = 0 \tag{14}$$

$$m_n - D_n / d_n = 0 \tag{15}$$

$$C_{mn} - [(4m_n - 1)/(4m_n - 4) + 0.615 m_n^{-1}] = 0 \tag{16}$$

$$K_n - G d_n^{\,4} / (8 n_n D_n^{\,3}) = 0 \tag{17}$$

$$H_{\min n} - (n_n + 1) d_n = 0 \tag{18}$$

$$N_n - (n_n + 1.5) = 0 \tag{19}$$

$$H_{0n} - [H_{\min n} + P_{vn}(1 + K_{vd}) / K_n] = 0 \tag{20}$$

$$m_w - D_w / d_w = 0 \tag{21}$$

$$C_{mw} - [(4m_w - 1)/(4m_w - 4) + 0.615 m_w^{-1}] = 0 \tag{22}$$

$$K_w - G d_w^{\ 4} / (8 n_w D_w^{\ 3}) = 0 \tag{23}$$

$$H_{\min w} - (n_w + 1) d_w = 0 \tag{24}$$

$$N_w - (n_w + 1.5) = 0 \tag{25}$$

$$k_2 = K_w + K_n \tag{26}$$

$$u = b z_5 \tag{27}$$

2) Knowledge in cybernetics can be expressed as

$$G_c(s) = K + T_I / s + T_D s \tag{28}$$

$$\dot{z}_6 = z_5 \tag{29}$$

$$T_D \dot{z}_6 = z_3 - z_1 - K z_6 - T_I z_5 \tag{30}$$

$$K < 100 \tag{31}$$

$$T_I, T_D < 30 \tag{32}$$

3) Knowledge in dynamics can be expressed as

$$\dot{z}_s = z_1 \tag{33}$$

$$\dot{z}_t = z_2 \tag{34}$$

$$\dot{z}_1 = [-k_1(z_s - z_t) - c(z_1 - z_2) + u] / m_1 \tag{35}$$

$$\dot{z}_2 = [k_1(z_s - z_t) + c(z_1 - z_2) - u + k_2(r - z_t)] / m_2 \tag{36}$$

$$r = 0.02\sin 14t \tag{37}$$

$$\dot{z}_{rs} = z_3 \tag{38}$$

$$\dot{z}_{rt} = z_4 \tag{39}$$

$$\dot{z}_3 = [-k_1(z_{rs} - z_{rt}) - c_{rs}z_3]/m_1 \tag{40}$$

$$\dot{z}_4 = [k_1(z_{rs} - z_{rt}) - c_{rt}z_4 + k_2(r - z_{rt})]/m_2 \tag{41}$$

4) Knowledge in other disciplinary can be expressed as

$$\ddot{z}_s < 0.3g \tag{42}$$

$$z_t - r < 0.02 \tag{43}$$

$$z_s - z_t < 0.05 \tag{44}$$

$$\dot{z}_s \le 0.00027 \cdot v + 0.030 \tag{45}$$

$$W = 2.7\sqrt[10]{0.325 \cdot z_s^3 \cdot f^7} \le 2.5 \tag{46}$$

$$D_x = c/2\sqrt{(k_1 + k_2)(m_1 + m_2)} \tag{47}$$

$$\tau_{\max n} = 8P_{\max n}D_nC_{mn}/\pi d_n^{\ 3} \le [\tau] \tag{48}$$

$$h_1 = H_{0n}/D_n - 3.5 \le 0 \tag{49}$$

$$h_2 = H_{0w}/D_w - 3.5 \le 0 \tag{50}$$

$$\tau_{\max w} = 8P_{\max w}D_wC_{mw}/\pi d_w^{\ 3} \le [\tau] \tag{51}$$

Step 2. Multidisciplinary parameter decision-making

Based on the above knowledge from multidisciplinary, the parameter decision-making model can be established by Eqs. (1). The initial intervals are shown as the second and third column in Table 1. The results of consistent interval are shown as the fourth and fifth column in Table 1.

Table 1. The design results of bogie dumping components

Variable name	Initial upper bound	Initial lower bound	Consistent upper bound	Consistent lower bound
$P_{\max n}$	30000	20000	28787	20000
D_n	0.15	0.1	0.15	0.142
d_n	0.03	0.025	0.029	0.025
C_{mn}	1.57	1.2	1.30	1.22
m_n	6	3	6	5.12
K_n	300000	250000	284470	265500
$H_{\min n}$	0.5	0.05	0.375	0.15
n_n	7	6	6.7	6.47
N_n	9	5	9	7.7
P_{vn}	18000	17000	18000	17130
K_{vd}	0.7	0.5	0.7	0.5
H_{0n}	0.4	0.2	0.4	0.31
$P_{\max w}$	55000	49000	53500	49000
D_w	0.25	0.22	0.25	0.22
d_w	0.050	0.038	0.045	0.038
C_{mw}	1.57	1.2	1.39	1.23
m_w	6	3	5.87	4.99
K_w	550000	400000	550000	478500
$H_{\min w}$	0.5	0.05	0.4354	0.2054
n_w	5	4	4.77	4.12
N_w	6	5	6	5.55
P_{vw}	40000	28000	36200	28620
H_{0w}	0.4	0.2	0.4	0.25
k_1	0.28	0.28	0.28	0.28
Dx	0.3	0.2	0.3	0.2
c	90	100	94.75	99.98
K	1	10	1.17	10
T_I	1	10	1.23	9.66
T_D	1	10	1.44	9.98

5 Conclusions

In this paper, a knowledge-based parameter decision-making method for multidisciplinary design is presented. In the process of multidisciplinary knowledge modeling, knowledge from multidisciplinary specification is formulated as a set of constraints. Multidisciplinary knowledge model can be used to detect conflict and obtain consistent solution space for multidisciplinary parameter design. A design example is proved that the proposed method is efficient. The method described in this paper was used to develop knowledge driven multidisciplinary design system, which is now running in a railway vehicle design project.

Acknowledgement

This paper is supported by the National Natural Science Foundation of China (Nos. 50575142 and 60304015) and the Shanghai Committee of Science and Technology (Nos. 055107048 and 04ZR14081).

References

1. Ahn, J., Kwon, J.H.: An efficient strategy for reliability-based multidisciplinary design optimization using BLISS. Structural and Multidisciplinary Optimization 5, 363–372 (2006)
2. Rosenman, M.A., Smith, G., Maher, M.L., Ding, L., Marchant, D.: Multidisciplinary collaborative design in virtual environments. Automation in Construction 1, 37–44 (2007)
3. Lee, J., Jeong, H.: A decomposition based design method coordinated by disciplinary subspace optimization. Mechanical Systems Machine Elements and Manufacturing 3, 935–941 (2006)
4. Chen, L., Jin, G.D.: Product modeling for multidisciplinary collaborative design. International Journal of Advanced Manufacturing Technology 7(8), 589–600 (2006)
5. Kong, S.H., Noh, S.D., Han, Y.G., Kim, G., Lee, K.I.: Internet-based collaboration system: Press-die design process for automobile manufacturer. International Journal of Advanced Manufacturing Technology 9, 701–708 (2002)
6. Yin, G.F., Tian, G.Y., Taylor, D.: A web-based remote cooperative design for spatial cam mechanisms. International Journal of Advanced Manufacturing Technology 8, 557–563 (2002)
7. Wang, Y.D., Shen, W.M., Ghenniwa, H.: A web/agent-based multidisciplinary design optimization environment. Computers in Industry 1, 17–28 (2003)
8. Gantois, K., Morris, A.J.: The multi-disciplinary design of a large-scale civil aircraft wing taking account of manufacturing costs. Structural and Multidisciplinary Optimization 1, 31–46 (2004)
9. Giassi, A., Bennis, F., Maisonneuve, J.J.: Multidisciplinary design optimisation and robust design approaches applied to concurrent design. Structural and Multidisciplinary Optimization 5, 356–371 (2004)
10. Bai, Y.W., Chen, Z.N., Bin, H.Z., Hu, J.: Collaborative design in product development based on product layout model. Robotics and Computer-Integrated Manufacturing 1, 55–65 (2005)

Resolution of Singularities and Stochastic Complexity of Complete Bipartite Graph-Type Spin Model in Bayesian Estimation

Miki Aoyagi and Sumio Watanabe

Precision and Intelligence Laboratory, Tokyo Institute of Technology,
4259 Nagatsuda, Midori-ku, R2-5, Yokohama, 226-8503, Japan
aoyagi.m.aa@m.titech.ac.jp, swatanab@pi.titech.ac.jp

Abstract. In this paper, we obtain the main term of the average stochastic complexity for certain complete bipartite graph-type spin models in Bayesian estimation. We study the Kullback function of the spin model by using a new method of eigenvalue analysis first and use a recursive blowing up process for obtaining the maximum pole of the zeta function which is defined by using the Kullback function. The papers [1,2] showed that the maximum pole of the zeta function gives the main term of the average stochastic complexity of the hierarchical learning model.

1 Introduction

The spin model in statistical physics is also called the Boltzmann machine. In mathematics, the spin model can be regarded as the Bayesian network or the graphical model. So, the model is widely used in many fields. However, its many theoretical problems have been unsolved so far. Clarifying its stochastic complexity is one of those problems in the artificial intelligence. Stochastic complexities are used in model selection methods well. Therefore, it is an important problem to know the behavior of stochastic complexities. The fact that the spin model is a non-regular statistical model makes the problem difficult. We cannot analyze it by using classic theories of regular statistical models, since their Fisher matrix functions are singular. This is the reason why we may not apply model selection methods such as AIC[3], TIC[4], HQ[5], NIC[6], BIC[7], MDL[8] to the non-regular statistical model.

Recently, the papers [1,2] showed that the maximum pole of the zeta function of hierarchical learning models gives the main term of their average stochastic complexity. The results are for all non-regular statistical models which include not only the spin model but also the layered neural network, the reduced rank regression and the normal mixture model. It is known that the desingularization of an arbitrary polynomial can be obtained by using a blowing up process (Hironaka's Theorem [9]). Therefore, the maximum pole is obtained by a blowing up process of its Kullback function.

However, in spite of such results, it is still difficult to obtain stochastic complexities by the following two main reasons. (1) The desingularization of any

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 443–454, 2007.

polynomial in general, although it is known as a finite process, is very difficult. Furthermore, most of the Kullback functions of non-regular statistical models are degenerate (over $\mathbb{R}$) with respect to their Newton polyhedrons, singularities of the Kullback functions are not isolated, and the Kullback functions are not simple polynomials, i.e., they have parameters. Therefore, to obtain the desingularization of the Kullback functions is a new problem even in mathematics, since these singularities are very complicated and so most of them have not been investigated so far. (2) Since the main purpose is for obtaining the maximum pole, getting the desingularization is not enough for us. We need some techniques for comparing poles. However, no theorems for comparing poles have developed as far as we know.

Therefore, the exact main terms of the average stochastic complexities of spin models were unknown, while upper bounds were reported in several papers [10,11]. In this paper, we clarify explicitly the main terms of the stochastic complexities of certain complete bipartite graph-type spin models, by using a new method of eigenvalue analysis and a recursive blowing up process (Theorem 4).

We already have obtained the exact main terms of the average stochastic complexities for the three layered neural network in [12] and [13], and the reduced rank regression in [14].

There are usually direct and inverse problems to be considered. The direct problem is to solve the stochastic complexity with a known true density function. The inverse problem is to find proper learning models and learning algorithms under the condition of an unknown true density function. The inverse problem is important for practical usage, but in order to solve the inverse problem, first the direct problem has to be solved. So it is necessary and crucial to construct fundamental mathematical theories for solving the direct problem. Our standpoint comes from that direct problem.

This paper consists of five sections. In Section 2, we summary Bayesian learning models [1,2]. Section 3 contains Hironaka's Theorem [9]. In Section 4, our main results are stated. In Section 5, we conclude our paper.

2 Bayesian Learning Models

Let $x^n := \{x_i\}_{i=1}^n$ be n training samples randomly selected from a true probability density function $q(x)$. Consider a learning model $p(x|w)$, where w is a parameter. We assume that the true probability density function $q(x)$ is defined by $q(x) = p(x|w^*)$, where w^* is constant.

Let $K_n(w) = \dfrac{1}{n}\displaystyle\sum_{i=1}^{n} \log \frac{p(x^n|w^*)}{p(x^n|w)}$.

The average stochastic complexity or the free energy is defined by

$$F(n) = -E_n\{\log \int \exp(-nK_n(w))\psi(w)dw\}.$$

Let $p(w|x^n)$ be the *a posteriori* probability density function:

$p(w|x^n) = \frac{1}{Z_n}\psi(w)\prod_{i=1}^{n} p(x_i|w)$, where $\psi(w)$ is an *a priori* probability density function on the parameter set W and $Z_n = \int_W \psi(w)\prod_{i=1}^{n} p(x_i|w)\mathrm{d}w$.

So the average inference $p(x|x^n)$ of the Bayesian density function is given by $p(x|x^n) = \int p(x|w)p(w|x^n)\mathrm{d}w$.

Set $K(q||p) = \sum_x q(x)\log\frac{q(x)}{p(x|x^n)}$. This function represents a measure function between the true density function $q(x)$ and the predictive density function $p(x|x^n)$. It always takes a positive value and satisfies $K(q||p) = 0$ if and only if $q(x) = p(x|x^n)$.

The generalization error $G(n)$ is its expectation value over training samples:

$$G(n) = E_n\{\sum_x p(x|w^*)\log\frac{p(x|w^*)}{p(x|x^n)}\},$$

which satisfies $G(n) = F(n+1) - F(n)$ if it has an asymptotic expansion.

Define the zeta function $J(z)$ of a complex variable z for the learning model by $J(z) = \int K(w)^z\psi(w)\mathrm{d}w$, where $K(w)$ is the Kullback function: $K(w) = \sum_x p(x|w^*)\log\frac{p(x|w^*)}{p(x|w)}$. Then, for the maximum pole $-\lambda$ of $J(z)$ and its order θ, we have

$$F(n) = \lambda\log n - (\theta-1)\log\log n + O(1), \tag{1}$$

where $O(1)$ is a bounded function of n, and

$$G(n) \cong \lambda/n - (\theta-1)/(n\log n) \text{ as } n\to\infty. \tag{2}$$

Therefore, our aim is to obtain λ and θ in this paper.

We state Lemmas 2 and 3 in [14] below which are frequently used in this paper. Define the norm of a matrix $C = (c_{ij})$ by $||C|| = \sqrt{\sum_{i,j}|c_{ij}|^2}$.

Lemma 1 ([14]). *Let U be a neighborhood of $w_0 \in \mathbb{R}^d$, $C(w)$ be an analytic $H\times H'$ matrix function from U, $\psi(w)$ be a C^∞ function from U with compact support, and P, Q be any regular $H\times H$, $H'\times H'$ matrices, respectively. Then the maximum pole of $\int_U ||C(w)||^{2z}\psi(w)dw$ is the same of $\int_U ||PC(w)Q||^{2z}\psi(w)dw$.*

3 Resolution of Singularities

In this section, we introduce Hironaka's Theorem [9] on a resolution of singularities and construction of blowing up. Blowing up is a main tool in a resolution of singularities of an algebraic variety.

Theorem 1 (Hironaka [9])
Let f be a real analytic function in a neighborhood of $w = (w_1,\cdots,w_d) \in \mathbb{R}^d$ with $f(w) = 0$. There exists an open set $V \ni w$, a real analytic manifold U and a proper analytic map μ from U to V such that

(1) $\mu : U - \mathcal{E} \to V - f^{-1}(0)$ *is an isomorphism, where* $\mathcal{E} = \mu^{-1}(f^{-1}(0))$*,*
(2) for each $u \in U$*, there is a local analytic coordinate system* $(u_1, \cdots, u_n)$ *such that* $f(\mu(u)) = \pm u_1^{s_1} u_2^{s_2} \cdots u_n^{s_n}$*, where* $s_1, \cdots, s_n$ *are non-negative integers.*

Next we explain about blowing up along a manifold used in this paper [15]. Define a manifold $\mathcal{M}$ by gluing k open sets $U_i \cong \mathbb{R}^d$, $i = 1, 2, \cdots, k (d \geq k)$ as follows. Denote a coordinate system of U_i by $(\xi_{1i}, \cdots, \xi_{di})$.

Define an equivalence relation $(\xi_{1i}, \xi_{2i}, \cdots, \xi_{di}) \sim (\xi_{1j}, \xi_{2j}, \cdots, \xi_{dj})$ at $\xi_{ji} \neq 0$ and $\xi_{ij} \neq 0$, by $\xi_{ij} = 1/\xi_{ji}, \xi_{jj} = \xi_{ii}\xi_{ji}, \xi_{hj} = \xi_{hi}/\xi_{ji} (1 \leq h \leq k, h \neq i, j), \xi_{\ell j} = \xi_{\ell i} (k+1 \leq \ell \leq d)$, and set $\mathcal{M} = \coprod_{i=1}^k U_i / \sim$. Also define $\pi : \mathcal{M} \to \mathbb{R}^d$ by $U_i \ni (\xi_{1i}, \cdots, \xi_{ni}); \mapsto (\xi_{ii}\xi_{1i}, \cdots, \xi_{ii}\xi_{i-1i}, \xi_{ii}, \xi_{ii}\xi_{i+1i}, \cdots, \xi_{ii}\xi_{ki}, \xi_{k+1i}, \cdots, \xi_{di})$.

This map is well-defined and called blowing up along

$$X = \{(w_1, \cdots, w_k, w_{k+1}, \cdots, w_d) \in \mathbb{R}^d \mid w_1 = \cdots = w_k = 0\}.$$

The blowing map satisfies (1) $\pi : \mathcal{M} \to \mathbb{R}^d$ is proper and (2) $\pi : \mathcal{M} - \pi^{-1}(X) \to \mathbb{R}^d - X$ is isomorphic.

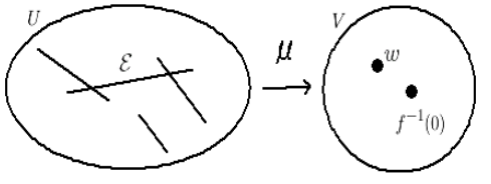


Fig. 1. Hironaka Theorem

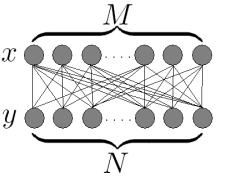


Fig. 2. A complete bipartite graph-type spin model

4 Spin Models

For simplicity, we use the notation $\mathrm{d}a$ instead of $\prod_{i=1}^{H} \prod_{j=1}^{H'} \mathrm{d}a_{ij}$ for $a = (a_{ij})$.

Let $2 \leq M \in \mathbb{N}$ and $N \in \mathbb{N}$. Consider a complete bipartite graph-type spin model

$$p(x, y|a) = \frac{\exp(\sum_{i=1}^{M} \sum_{j=1}^{N} a_{ij} x_i y_j)}{Z(a)}, Z(a) = \sum_{x_i = \pm 1, y_i = \pm 1,} \exp(\sum_{i=1}^{M} \sum_{j=1}^{N} a_{ij} x_i y_j),$$

with $x = (x_j) \in \{1, -1\}^M$ and $y = (y_j) \in \{1, -1\}^N$.

We have

$$p(x|a) = \frac{\prod_{j=1}^{N} (\prod_{i=1}^{M} \exp(a_{ij} x_i) + \prod_{i=1}^{M} \exp(-a_{ij} x_i))}{Z(a)}$$

$$= \{\prod_{j=1}^{N} (\prod_{i=1}^{M} (1 + x_i \tanh(a_{ij})) + \prod_{i=1}^{M} (1 - x_i \tanh(a_{ij})))\} \frac{\prod_{j=1}^{N} \prod_{i=1}^{M} \cosh(a_{ij})}{Z(a)}$$

$$= \frac{\prod_{j=1}^{N} \prod_{i=1}^{M} \cosh(a_{ij})}{Z(a)}$$

$$\times \prod_{j=1}^{N} (2 \sum_{0 \leq p \leq M/2} \sum_{i_1 < \cdots < i_{2p}} x_{i_1} x_{i_2} \cdots x_{i_{2p}} \tanh(a_{i_1 j}) \tanh(a_{i_2 j}) \cdots \tanh(a_{i_{2p} j})).$$

Let $B = (b_{ij}) = (\tanh(a_{ij}))$.

Denote $B^J = \prod_{i=1}^{M}\prod_{j=1}^{N} b_{ij}^{J_{ij}}$ and $x^J = \prod_{i=1}^{M} x_i^{\sum_{j=1}^{N} J_{ij}}$, where $J = (J_{ij})$ is an $M \times N$ matrix with $J_{ij} \in \{0, 1\}$.

Then we have

$$p(x|a) = \frac{2^N \prod_{j=1}^{N}\prod_{i=1}^{M} \cosh(a_{ij})}{Z(a)} \sum_{J:\sum_{i=1}^{M} J_{ij} \text{ even for all j}} B^J x^J.$$

Let $Z(b) = \dfrac{Z(a)}{2^N \prod_{j=1}^{N}\prod_{i=1}^{M} \cosh(a_{ij})}$. Set $\mathcal{I} = \{I \in \{0,1\}^M | \sum_{i=1}^{M} I_i \text{ is even }\}$, and $B^I = \sum_{\substack{J:\sum_{i=1}^{M} J_{ij} \text{ is even} \\ \sum_{j=1}^{N} J_{ij} = I_i \mod 2}} B^J$ for $I \in \mathcal{I}$. Then we have $p(x|a) = \dfrac{1}{Z(b)} \sum_{I\in\mathcal{I}} B^I x^I$ and $Z(b) = 2^N B^0$. Since $\sum_{0\le i\le M/2} \binom{M}{2i} = ((1+1)^M + (1-1)^M)/2 = 2^{M-1}$, the number of all elements in $\mathcal{I}$ is 2^{M-1}.

Assume that a true distribution is $p(x|a^*)$ with $a^* = (a_{ij}^*)$. Then the Kullback function $K(a)$ is

$$\sum_{x_i=\pm 1} p(x|a^*)(\log p(x|a^*) - \log p(x|a)) = \sum_{x_i=\pm 1} p(x|a^*) \sum_{i=2}^{\infty} \frac{(-1)^i}{i} (\frac{p(x|a)}{p(x|a^*)} - 1)^i.$$

Since we consider a neighborhood of $\frac{p(x|a)}{p(x|a^*)} = 1$, we only need to obtain the maximum pole of $J(z) = \int \Psi_0^z \mathrm{d}b$, where

$$\Psi_0 = \sum_{x_i=\pm 1} \frac{(p(x|a) - p(x|a^*))^2}{p(x|a^*)} = \sum_{x_i=\pm 1} \frac{(\frac{\sum_{I\in\mathcal{I}} B^I x^I}{Z(b)} - \frac{\sum_{I\in\mathcal{I}} B^{*I} x^I}{Z(b^*)})^2}{p(x|a^*)}.$$

By Lemma 5 in [1], we may replace Ψ_0 by

$$\Psi_1 = \sum_{I\in\{0,1\}^M} 2^{2N} (\frac{B^I}{Z(b)} - \frac{B^{*I}}{Z(b^*)})^2 = \sum_{I\in\{0,1\}^M} (\frac{B^I}{B^0} - \frac{B^{*I}}{B^{*0}})^2.$$

Assume that the true distribution is $p(x|a^*)$ with $a^* = 0$. By using Lemma 1, Ψ_1 can be replaced by

$$\Psi(b) = \sum_{I\neq 0\in\mathcal{I}} (B^I)^2, \tag{3}$$

and from now on, we consider the zeta function $J(z) = \int_V \Psi^z \mathrm{d}b$, where V is a sufficiently small neighborhood of 0.

Let $I, I', I'' \in \mathcal{I}$. We set $B_N^I = B^I$ and $b_j^I = \prod_{i=1}^{M} b_{ij}^{I_i}$. Also set

$$B_N = (B_N^I) = (B_N^{(0,\ldots,0)}, B_N^{(1,1,0,\ldots,0)}, B_N^{(1,0,1,0,\ldots,0)}, \ldots).$$

We have $B_N^I = \sum_{I'+I''=I \mod 2} b_N^{I''} B_{N-1}^{I'}$.

Now consider the eigenvalues of the matrix $C_N = (c_N^{I,I'})$ where $c_N^{I,I'} = b_N^{I''}$ with $I' + I'' = I \mod 2$. Note that $B_N = C_N B_{N-1}$. Let $\ell = (\ell_1, \ldots, \ell_{2^{M-1}}) = (\ell_I) \in \{-1, 1\}^{2^{M-1}}$ with $\ell_{(0,\ldots,0)} = 1$. ℓ is an eigenvector, if and only if $\sum_{I'\in\mathcal{I}} c_N^{I,I'} \ell_{I'} = \ell_I \sum_{I'\in\mathcal{I}} c_N^{(0,\ldots,0),I'} \ell_{I'} = \ell_I \sum_{I'\in\mathcal{I}} b_N^{I'} \ell_{I'}$. That is,

ℓ is an eigenvector $\Longleftrightarrow$ if $I + I' = I'' \mod 2$ $(I + I' + I'' = 0 \mod 2)$ then $\ell_{I''} = \ell_I \ell_{I'}$ $(\ \ell_I \ell_{I'} \ell_{I''} = 1)$.

Denote the number of all elements in a set K by $\#K$.

Theorem 2. *Let $K_1, K_2 \subset \{1, \ldots, M\}$, $1 \in K_2$, $K_1 \cap K_2 = \phi$, and $K_1 \cup K_2 = \{1, \ldots, M\}$.*

Set $\ell_I = \begin{cases} -1, & \text{if } \#\{i \in K_1 : I_i = 1\} \text{ is odd,} \\ 1, & \text{otherwise.} \end{cases}$ *If $K_1 = \phi$, set $\ell = (1, \ldots, 1)$.*

Then $\ell = (\ell_I)$ is an eigenvector of C_N and its eigenvalue is $\sum_{I \in \mathcal{I}} \ell_I b_N^I$.

Proof. Assume that $I' + I'' + I''' = 0 \mod 2$. If all $\#\{i \in K_1 : I'_i = 1\}$, $\#\{i \in K_1 : I''_i = 1\}$ and $\#\{i \in K_1 : I'''_i = 1\}$ are even, then $\ell_{I'} \ell_{I''} \ell_{I'''} = 1$.

If $\#\{i \in K_1 : I'_i = 1\}$ and $\#\{i \in K_1 : I''_i = 1\}$ are odd, then $\#\{i \in K_1 : I'''_i = 1\}$ is even and $\ell_{I'} \ell_{I''} \ell_{I'''} = 1$ since $I' + I'' + I''' = 0 \mod 2$.

If $\#\{i \in K_1 : I'_i = 1\}$ is odd, then $\#\{i \in K_1 : I''_i = 1\}$ or $\#\{i \in K_1 : I'''_i = 1\}$ is odd, since $I' + I'' + I''' = 0 \mod 2$. □

Since we have 2^{M-1} pairs of K_1, K_2 with $1 \in K_2$, $K_1 \cap K_2 = \phi$ and $K_1 \cup K_2 = \{1, \ldots, M\}$, those eigenvectors ℓ's span the whole space $\mathbb{R}^{2^{M-1}}$ and are orthogonal to each other.

Set $\mathbf{1} = (1, \ldots, 1)^t \in \mathbb{Z}^{2^{M-1}-1}$ (t denotes the transpose). Let D be the matrix by arranging the eigenvectors ℓ's such that $D = \begin{pmatrix} 1 & \mathbf{1}^t \\ \mathbf{1} & D' \end{pmatrix}$ and $DD = 2^{M-1} E$, where E is the unit matrix.

Since $DD = \begin{pmatrix} 2^{M-1} & \mathbf{1}^t D' \\ \mathbf{1} + D'\mathbf{1} & \mathbf{1}\mathbf{1}^t + D'D' \end{pmatrix} = 2^{M-1} E$, we have $D'\mathbf{1} = -\mathbf{1}$.

Theorem 3. *Let $C'_j = DC_j D / 2^{M-1} = DC_j D^{-1} = \begin{pmatrix} s_{0j} & 0 & 0 \cdots & 0 \\ 0 & s_{1j} & 0 \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 \cdots & s_{2^{M-1}-1,j} \end{pmatrix}$ which is the diagonal matrix. We have the followings.*

(1) *Let* $d_{ij} = \begin{cases} 1, & \text{if } i = 1 \text{ or } j = 1, \\ D^{I,J}, & \text{if } I = (1, 0, \ldots, 0, \overset{i}{1}, 0, \ldots, 0) \\ & \text{and } J = (1, 0, \ldots, 0, \overset{j}{1}, 0, \ldots, 0). \end{cases}$

Then $D^{I,J} = \prod_{i \in I, j \in J} d_{ij}$ for all $I, J \in \mathcal{I}$.

(2) $B_N = C_N B_{N-1} = C_N \cdots C_2 B_1 = DC'_N \cdots C'_2 D^{-1} B_1 = \dfrac{DC'_N \cdots C'_1 \mathbf{1}}{2^{M-1}}$.

(3) *We have $2^{M-1} D'^{-1} = D' - \mathbf{1}\mathbf{1}^t$.*

(4) *Let $\tilde{B}_1 = (B_1^I)_{I \neq 0}$, $\tilde{B}_N = (B_N^I)_{I \neq 0}$ and* $S = (\prod_{j=2}^N s_{0j}) \begin{pmatrix} 1 \cdots 1 \\ \vdots \ \ \vdots \\ 1 \cdots 1 \end{pmatrix} + \begin{pmatrix} \prod_{j=2}^N s_{1j} & 0 & 0 \cdots & 0 \\ 0 & \prod_{j=2}^N s_{2j} & 0 \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 \cdots & \prod_{j=2}^N s_{2^{M-1}-1,j} \end{pmatrix}$.

We have

$$(\det S)D'^{-1}S^{-1}D'^{-1}2^{M-1}\tilde{B}_N = (\det S)\tilde{B}_1 - (\mathbf{1}\ D')\begin{pmatrix} \prod_{i\neq 0}\prod_{j=2}^N s_{ij} \\ \prod_{i\neq 1}\prod_{j=2}^N s_{ij} \\ \vdots \\ \prod_{i\neq 2^{M-1}-1}\prod_{j=2}^N s_{ij} \end{pmatrix}.$$

(5) *The corresponding element to* I *of* $(\mathbf{1}\ D')\begin{pmatrix} \prod_{i\neq 0}\prod_{j=2}^N s_{ij} \\ \prod_{i\neq 1}\prod_{j=2}^N s_{ij} \\ \vdots \\ \prod_{i\neq 2^{M-1}-1}\prod_{j=2}^N s_{ij} \end{pmatrix}$ *consists of monomials* $c_J\prod_{i=1}^M\prod_{j=2}^N b_{ij}^{J_{ij}}$, *where* $c_J \in \mathbb{R}$, $0 \leq J_{ij} \in \mathbb{Z}$ *and* $\sum_{j=1}^N J_{ij} = I_i \mod 2$.

Proof. (5) is obtained by

$$(C_N\cdots C_2)^{-1} = D\begin{pmatrix} 1/\prod_{j=2}^N s_{0j} & 0 & \cdots & 0 \\ 0 & 1/\prod_{j=2}^N s_{1j} & \cdots & 0 \\ \vdots & & \vdots & \\ 0 & 0 & \cdots & 1/\prod_{j=2}^N s_{2^{M-1}j} \end{pmatrix} D^{-1}.$$

We prove only (4). Let $H = 2^{M-1}-1$. We have

$$\begin{aligned} 2^{M-1}\tilde{B}_N &= (\mathbf{1}\ D')C'_N\cdots C'_2\begin{pmatrix} 1 & \mathbf{1}^t \\ \mathbf{1} & D' \end{pmatrix}B_1 \\ &= (\mathbf{1}\ D')\begin{pmatrix} \prod_{j=2}^N s_{0j} & 0 & 0\cdots & 0 \\ 0 & \prod_{j=2}^N s_{1j} & 0\cdots & 0 \\ \vdots & \vdots & \vdots\cdots & \vdots \\ 0 & 0 & 0\cdots & \prod_{j=2}^N s_{H,j} \end{pmatrix}\begin{pmatrix} 1 & \mathbf{1}^t \\ \mathbf{1} & D' \end{pmatrix}B_1 \\ &= (\mathbf{1}\ D')\left\{\begin{pmatrix} \prod_{j=2}^N s_{0j} \\ \prod_{j=2}^N s_{1j} \\ \vdots \\ \prod_{j=2}^N s_{H,j} \end{pmatrix} + \begin{pmatrix} \prod_{j=2}^N s_{0j} & 0 & 0\cdots & 0 \\ 0 & \prod_{j=2}^N s_{1j} & 0\cdots & 0 \\ \vdots & \vdots & \vdots\cdots & \vdots \\ 0 & 0 & 0\cdots & \prod_{j=2}^N s_{H,j} \end{pmatrix}\begin{pmatrix} \mathbf{1}^t \\ D' \end{pmatrix}\tilde{B}_1\right\} \\ &= (\mathbf{1}\ D')\begin{pmatrix} \prod_{j=2}^N s_{0j} \\ \prod_{j=2}^N s_{1j} \\ \vdots \\ \prod_{j=2}^N s_{H,j} \end{pmatrix} + D'(-\mathbf{1}\ E)\begin{pmatrix} \prod_{j=2}^N s_{0j} & 0 & 0\cdots & 0 \\ 0 & \prod_{j=2}^N s_{1j} & 0\cdots & 0 \\ \vdots & \vdots & \vdots\cdots & \vdots \\ 0 & 0 & 0\cdots & \prod_{j=2}^N s_{H,j} \end{pmatrix}\begin{pmatrix} -\mathbf{1}^t \\ E \end{pmatrix}D'\tilde{B}_1 \\ &= (\mathbf{1}\ D')\begin{pmatrix} \prod_{j=2}^N s_{0j} \\ \prod_{j=2}^N s_{1j} \\ \vdots \\ \prod_{j=2}^N s_{H,j} \end{pmatrix} + D'SD'\tilde{B}_1. \end{aligned}$$

Therefore $D'^{-1}2^{M-1}\tilde{B}_N = \left(-\mathbf{1}\ E\right)\begin{pmatrix} \prod_{j=2}^N s_{0j} \\ \prod_{j=2}^N s_{1j} \\ \vdots \\ \prod_{j=2}^N s_{H,j} \end{pmatrix} + SD'\tilde{B}_1.$

We have

$$S^{-1}_{i_1 j_1} = (\det S)^{-1}\begin{cases} \sum_{i_2=0, i_2\neq i_1}^{H} \prod_{0\le i\le H, i\neq i_1, i_2} \prod_{j=2}^N s_{ij}, & \text{if } i_1 = j_1, \\ -\prod_{0\le i\le H, i\neq i_1, j_1} \prod_{j=2}^N s_{ij}, & \text{if } i_1 \neq j_1, \end{cases}$$

and $\det S = \sum_{i_2=0}^{H} \prod_{i\neq i_2} \prod_{j=2}^N s_{ij}$.

Let $\mathbf{s} = \begin{pmatrix} \prod_{i\neq 0} \prod_{j=2}^N s_{ij} \\ \prod_{i\neq 1} \prod_{j=2}^N s_{ij} \\ \vdots \\ \prod_{i\neq H} \prod_{j=2}^N s_{ij} \end{pmatrix}$ and $\tilde{\mathbf{s}} = \begin{pmatrix} \prod_{i\neq 1} \prod_{j=2}^N s_{ij} \\ \prod_{i\neq 2} \prod_{j=2}^N s_{ij} \\ \vdots \\ \prod_{i\neq H} \prod_{j=2}^N s_{ij} \end{pmatrix}$.

Since $(\det S)S^{-1}\begin{pmatrix} \prod_{j=2}^N s_{1j} - \prod_{j=2}^N s_{0j} \\ \vdots \\ \prod_{j=2}^N s_{H,j} - \prod_{j=2}^N s_{0j} \end{pmatrix} = \sum_{i_2=0}^{H} \prod_{i\neq i_2} \prod_{j=2}^N s_{ij}\mathbf{1} - 2^{M-1}\tilde{\mathbf{s}}$, we have

$$\begin{aligned} (\det S)D'^{-1}S^{-1}D'^{-1}2^{M-1}\tilde{B}_N &= (\det S)\tilde{B}_1 - \sum_{i_2=0}^{H}\prod_{i\neq i_2}\prod_{j=2}^{N} s_{ij}\mathbf{1} - 2^{M-1}D'^{-1}\tilde{\mathbf{s}} \\ &= (\det S)\tilde{B}_1 - \sum_{i_2=0}^{H}\prod_{i\neq i_2}\prod_{j=2}^{N} s_{ij}\mathbf{1} - (D' - \mathbf{1}\mathbf{1}^t)\tilde{\mathbf{s}} \\ &= (\det S)\tilde{B}_1 - \prod_{i\neq 0}\prod_{j=2}^{N} s_{ij}\mathbf{1} - D'\tilde{\mathbf{s}} = (\det S)\tilde{B}_1 - (\mathbf{1}\ D')\mathbf{s}, \end{aligned}$$

by using (3) $2^{M-1}D'^{-1} = D' - \mathbf{1}\mathbf{1}^t$. □

Theorem 4. *The average stochastic complexity $F(n)$ in (1) and the generalization error $G(n)$ in (2) are given by using the following maximum pole $-\lambda$ of $J(z)$ and its order θ.*

(Case 1)*: If $N = 1$ then $\lambda = M/4$ and $\theta = \begin{cases} 2, & \text{if } M = 2, \\ 1, & \text{if } M \ge 3. \end{cases}$*

(Case 2)*: If $M = 2$ then $\lambda = 1/2$ and $\theta = \begin{cases} 2, & \text{if } N = 1, \\ 1, & \text{if } N \ge 2. \end{cases}$*

(Case 3)*: If $M = 3$ then $\lambda = \begin{cases} 3/4, & \text{if } N = 1, \\ 3/2, & \text{if } N \ge 2, \end{cases}$ and $\theta = \begin{cases} 1, & \text{if } N = 1, \\ 3, & \text{if } N = 2, \\ 1, & \text{if } N \ge 3. \end{cases}$*

(Case 4)*: If $M = 4$ then $\lambda = \begin{cases} 1, & \text{if } N = 1, \\ 2, & \text{if } N = 2, \end{cases}$ and $\theta = 1$, if $N = 1, 2$.*

Proof. By Theorem 3 (4) and Lemma 1, we only need to consider the maximum pole of $J(z) = \int ||\Psi'||^{2z} \mathrm{d}b$, where $\Psi' = (\det S)\tilde{B}_1 - (\mathbf{1}\ D')\begin{pmatrix} \prod_{i\neq 0}\prod_{j=2}^{N} s_{ij} \\ \prod_{i\neq 1}\prod_{j=2}^{N} s_{ij} \\ \vdots \\ \prod_{i\neq H}\prod_{j=2}^{N} s_{ij} \end{pmatrix}$.

(Case 1): Since $B_1^I = \prod_{j\in I} b_{1j}$, we have the poles $-\dfrac{M}{4}$ and $-\dfrac{M-1}{2}$.

(Case 2): The fact $B^{11} = \sum_{k=1}^{N} b_{1k}b_{2k}(1+\cdots)$ yields Case 2.

(Case 3): Assume that $M = 3$.

Let $N \geq 2$. We have $D' = \begin{pmatrix} 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{pmatrix}$, $\begin{cases} s_{0j} = 1 + b_{1j}b_{2j} + b_{1j}b_{3j} + b_{2j}b_{3j}, \\ s_{1j} = 1 + b_{1j}b_{2j} - b_{1j}b_{3j} - b_{2j}b_{3j}, \\ s_{2j} = 1 - b_{1j}b_{2j} + b_{1j}b_{3j} - b_{2j}b_{3j}, \\ s_{3j} = 1 - b_{1j}b_{2j} - b_{1j}b_{3j} + b_{2j}b_{3j}, \end{cases}$

and $\Psi' = (\det S)\begin{pmatrix} b_{11}b_{21} \\ b_{11}b_{31} \\ b_{21}b_{31} \end{pmatrix} - (1, D')\begin{pmatrix} \prod_{i\neq 0}\prod_{j=2}^{N} s_{ij} \\ \prod_{i\neq 1}\prod_{j=2}^{N} s_{ij} \\ \prod_{i\neq 2}\prod_{j=2}^{N} s_{ij} \\ \prod_{i\neq 3}\prod_{j=2}^{N} s_{ij} \end{pmatrix}$.

Construct blowing up of Ψ' along the submanifold $\{b_{ij} = 0, 1 \leq i \leq M, 1 \leq j \leq N\}$.

Let $b_{11} = u$, $b_{ij} = ub'_{ij}$ for $(i,j) \neq (1,1)$.

Remark. By setting the general case as $b_{i_0j_0} = b'_{i_0j_0}$, $b_{ij} = b'_{i_0j_0}b'_{ij}$ for $(i,j) \neq (i_0, j_0)$, we have a manifold $\mathcal{M}$ by gluing MN open sets $U_{i_0j_0}$ with a coordinate system $(b'_{11}, b'_{12}, \cdots, b'_{MN})$ (cf. Section 3). We don't need to consider all cases since we obtain the same poles in $U_{i_0j_0}$ as those in U_{11}.

We have $\Psi'' = u^2(\det S)\begin{pmatrix} b'_{21} \\ b'_{31} \\ b'_{21}b'_{31} \end{pmatrix} + 4u^2\begin{pmatrix} \sum_{k=2}^{N} b'_{1k}b'_{2k} + u^2 f_1 \\ \sum_{k=2}^{N} b'_{1k}b'_{3k} + u^2 f_2 \\ \sum_{k=2}^{N} b'_{2k}b'_{3k} + u^2 f_3 \end{pmatrix}$, where f_1, f_2 and f_3 are polynomials of b'_{ij} with at least two degree.

By putting $\begin{pmatrix} b''_{21} \\ b''_{31} \end{pmatrix} = \begin{pmatrix} b'_{21} \\ b'_{31} \end{pmatrix} + 4\begin{pmatrix} \sum_{k=2}^{N} b'_{1k}b'_{2k} + u^2 f_1 \\ \sum_{k=2}^{N} b'_{1k}b'_{3k} + u^2 f_2 \end{pmatrix}/(\det S)$, we have

$$\Psi'' = \frac{u^2}{\det S} \times \begin{pmatrix} (\det S)^2 b''_{21} \\ (\det S)^2 b''_{31} \\ (b''_{21}\det S - 4\sum_{k=2}^{N} b'_{1k}b'_{2k} - 4u^2 f_1)(b''_{31}\det S - 4\sum_{k=2}^{N} b'_{1k}b'_{3k} - 4u^2 f_2) \end{pmatrix} + u^2\begin{pmatrix} 0 \\ 0 \\ 4\sum_{k=2}^{N} b'_{2k}b'_{3k} + 4u^2 f_3 \end{pmatrix}.$$

By using Lemma 1 again, the maximum pole of $\int ||\Psi''||^{2z} u^{3N} db$ is that of $J(z) = \int ||\Psi'''||^{2z} u^{3N} \mathrm{d}b$, where $\Psi''' = u^2 \begin{pmatrix} b''_{21} \\ b''_{31} \\ g_1 \end{pmatrix}$, and

$g_1 = (\sum_{k=2}^N b'_{1k} b'_{2k} + u^2 f_1)(\sum_{k=2}^N b'_{1k} b'_{3k} + u^2 f_2) + \frac{\det S}{4}(\sum_{k=2}^N b'_{2k} b'_{3k} + u^2 f_3)$.

Construct blowing up of Ψ''' along the submanifold $\{b''_{21} = 0, b''_{31} = 0, b'_{3k} = 0, 2 \le k \le N\}$. Then we have (I), (II) cases.

(I) Let $b'_{32} = v$, $b''_{21} = v b'''_{21}$, $b''_{31} = v b'''_{21}$, $b'_{3k} = v b''_{3k}$, for $3 \le k \le N$. Then $\Psi''' = u^2 v \begin{pmatrix} b'''_{21} \\ b'''_{31} \\ g'_1 \end{pmatrix}$, where $g'_1 = (\sum_{k=2}^N b'_{1k} b'_{2k} + u^2 f_1)(b'_{12} + \sum_{k=3}^N b'_{1k} b''_{3k} + u^2 f_2/v) + \frac{\det S}{4}(b'_{22} + \sum_{k=3}^N b'_{2k} b''_{3k} + u^2 f_3/v)$.

By Theorem 3 (5), we can set $f_2 = v f'_2$ and $f_3 = v f'_3$, where f'_2 and f'_3 are polynomials.

We have $(\sum_{k=2}^N b'_{1k} b'_{2k})(b'_{12} + \sum_{k=3}^N b'_{1k} b''_{3k}) + \frac{\det S}{4}(b'_{22} + \sum_{k=3}^N b'_{2k} b''_{3k})$

$$= (b'_{2,2}, b'_{2,3}, \cdots, b'_{2,N}) \left(\begin{pmatrix} b'_{1,2} \\ b'_{1,3} \\ \vdots \\ b'_{1,N} \end{pmatrix} (b'_{1,2}, b'_{1,3}, \cdots, b'_{1,N}) + \frac{\det S}{4} E \right) \begin{pmatrix} 1 \\ b''_{3,3} \\ \vdots \\ b''_{3,N} \end{pmatrix}.$$

Since $\begin{pmatrix} b'_{1,2} \\ b'_{1,3} \\ \vdots \\ b'_{1,N} \end{pmatrix} (b'_{1,2}, b'_{1,3}, \cdots, b'_{1,N}) + \frac{\det S}{4} E$ is regular, we can change variables from $(b'_{2,2}, b'_{2,3}, \cdots, b'_{2,N})$ to $(b''_{2,2}, b''_{2,3}, \cdots, b''_{2,N})$ by $(b''_{2,2}, b''_{2,3}, \cdots, b''_{2,N}) = (b'_{2,2}, b'_{2,3}, \cdots, b'_{2,N}) \left(\begin{pmatrix} b'_{1,2} \\ b'_{1,3} \\ \vdots \\ b'_{1,N} \end{pmatrix} (b'_{1,2}, b'_{1,3}, \cdots, b'_{1,N}) + \frac{\det S}{4} E \right)$. Moreover, let $b'''_{22} = b''_{2,2} + b''_{2,3} b''_{3,3} + \cdots + b''_{2,N} b''_{3,N}$.

Then, we have

$$\Psi''' = u^2 v \begin{pmatrix} b'''_{21} \\ b'''_{31} \\ b'''_{22} + u^2 f_4 \end{pmatrix},$$

where f_4 is a polynomial. Therefore, we have the poles $-\frac{3N}{4}, -\frac{N+1}{2}, -\frac{3}{2}$.

(II) Let $b''_{21} = v$, $b''_{31} = v b'''_{21}$, $b'_{3k} = v b''_{3k}$, for $2 \le k \le N$. Then we have the poles $-\frac{3N}{4}, -\frac{N+1}{2}$.

(Case 4): Let $M = 4$. We have $D' = \begin{pmatrix} -1 & -1 & -1 & 1 & 1 & 1 & -1 \\ -1 & 1 & 1 & -1 & -1 & 1 & -1 \\ -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & -1 \\ 1 & 1 & -1 & 1 & -1 & -1 & -1 \\ -1 & -1 & 1 & 1 & -1 & -1 & 1 \end{pmatrix}$ and

$$
\begin{aligned}
s_{0j} &= 1 + b_{1j}b_{2j} + b_{1j}b_{3j} + b_{1j}b_{4j} + b_{2j}b_{3j} + b_{2j}b_{4j} + b_{3j}b_{4j} + b_{1j}b_{2j}b_{3j}b_{4j}, \\
s_{1j} &= 1 + b_{2j}b_{3j} + b_{2j}b_{4j} + b_{3j}b_{4j} - b_{1j}(b_{2j} + b_{3j} + b_{4j} + b_{2j}b_{3j}b_{4j}), \\
s_{2j} &= 1 + b_{1j}b_{3j} + b_{1j}b_{4j} + b_{3j}b_{4j} - b_{2j}(b_{1j} + b_{3j} + b_{4j} + b_{1j}b_{3j}b_{4j}), \\
s_{3j} &= 1 + b_{1j}b_{3j} + b_{2j}b_{4j} + b_{1j}b_{2j}b_{3j}b_{4j} - (b_{1j} + b_{3j})(b_{2j} + b_{4j}), \\
s_{4j} &= 1 + b_{1j}b_{2j} + b_{3j}b_{4j} + b_{1j}b_{2j}b_{3j}b_{4j} - (b_{1j} + b_{2j})(b_{3j} + b_{4j}), \\
s_{5j} &= 1 + b_{1j}b_{2j} + b_{1j}b_{4j} + b_{2j}b_{4j} - b_{3j}(b_{1j} + b_{2j} + b_{4j} + b_{1j}b_{2j}b_{4j}), \\
s_{6j} &= 1 + b_{1j}b_{2j} + b_{1j}b_{3j} + b_{2j}b_{3j} - b_{4j}(b_{1j} + b_{2j} + b_{3j} + b_{1j}b_{2j}b_{3j}), \\
s_{7j} &= 1 + b_{1j}b_{4j} + b_{2j}b_{3j} + b_{1j}b_{2j}b_{3j}b_{4j} - (b_{1j} + b_{4j})(b_{2j} + b_{3j}).
\end{aligned}
$$

Let $M = 4$ and $N = 2$. Then we have

$$
\Psi' = \det S \begin{pmatrix} b_{11}b_{21} \\ b_{11}b_{31} \\ b_{11}b_{41} \\ b_{21}b_{31} \\ b_{21}b_{41} \\ b_{31}b_{41} \\ b_{11}b_{21}b_{31}b_{41} \end{pmatrix} - \begin{pmatrix} -b_{12}b_{22}(8 + f_1) \\ -b_{12}b_{32}(8 + f_2) \\ -b_{12}b_{42}(8 + f_3) \\ -b_{22}b_{32}(8 + f_4) \\ -b_{22}b_{42}(8 + f_5) \\ -b_{32}b_{42}(8 + f_6) \\ b_{12}b_{22}b_{32}b_{42}(40 + f_7) \end{pmatrix},
$$

where f_i's are polynomials of b_{ij} with at least two degree. As space is limited, we will omit the proof in detail, but we have the poles $-\frac{8}{4}, -\frac{6}{2}, -\frac{5}{2}, -\frac{9}{4}$. □

5 Conclusion

In this paper, we obtain the main term of the average stochastic complexity for certain complete bipartite graph-type spin models in Bayesian estimation (Theorem 4). We use a new method of eigenvalue analysis and a recursive blowing up method in algebraic geometry and show that these are effective for solving the problems in the artificial intelligence. Our future purpose is to improve our methods and apply them to more general cases. Since eigenvalue analysis can be applied to general cases, we seem to formulate a new direction for solving the behavior of the spin model's stochastic complexity.

The applications of our results are as follows. The explicit values of generalization errors have been used to construct mathematical foundation for analyzing and developing the precision of the MCMC method [16]. Moreover, these values have been compared to such as the generalization error of localized Bayes estimation [17].

Acknowledgments. This research was supported by the Ministry of Education, Science, Sports and Culture in Japan, Grant-in-Aid for Scientific Research 18079007 and 16700218.

References

1. Watanabe, S.: Algebraic analysis for nonidentifiable learning machines. Neural Computation 13(4), 899–933 (2001)
2. Watanabe, S.: Algebraic geometrical methods for hierarchical learning machines. Neural Networks 14(8), 1049–1060 (2001)
3. Akaike, H.: A new look at the statistical model identification. IEEE Trans. on Automatic Control 19, 716–723 (1974)
4. Takeuchi, K.: Distribution of an information statistic and the criterion for the optimal model. Mathematical Science 153, 12–18 (1976)
5. Hannan, E.J., Quinn, B.G.: The determination of the order of an autoregression. Journal of Royal Statistical Society Series B 41, 190–195 (1979)
6. Murata, N.J., Yoshizawa, S.G., Amari, S.: Network information criterion - determining the number of hidden units for an artificial neural network model. IEEE Trans. on Neural Networks 5(6), 865–872 (1994)
7. Schwarz, G.: Estimating the dimension of a model. Annals of Statistics 6(2), 461–464 (1978)
8. Rissanen, J.: Universal coding, information, prediction, and estimation. IEEE Trans. on Information Theory 30(4), 629–636 (1984)
9. Hironaka, H.: Resolution of Singularities of an algebraic variety over a field of characteristic zero. Annals of Math. 79, 109–326 (1964)
10. Yamazaki, K., Watanabe, S.: Singularities in Complete Bipartite Graph-Type Boltzmann Machines and Upper Bounds of Stochastic Complexities. IEEE Trans. on Neural Networks 16(2), 312–324 (2005)
11. Nishiyama, Y., Watanabe, S.: Asymptotic Behavior of Free Energy of General Boltzmann Machines in Mean Field Approximation. Technical report of IEICE NC2006 38, 1–6 (2006)
12. Aoyagi, M., Watanabe, S.: Resolution of Singularities and the Generalization Error with Bayesian Estimation for Layered Neural Network. IEICE Trans. J88-D-II,10, 2112–2124 (2005) (English version: Systems and Computers in Japan John Wiley & Sons Inc. 2005) (in press)
13. Aoyagi, M.: The zeta function of learning theory and generalization error of three layered neural perceptron. RIMS Kokyuroku, Recent Topics on Real and Complex Singularities 2006 (in press)
14. Aoyagi, M., Watanabe, S.: Stochastic Complexities of Reduced Rank Regression in Bayesian Estimation. Neural Networks 18, 924–933 (2005)
15. Watanabe, S., Hagiwara, K., Akaho, S., Motomura, Y., Fukumizu, K., Okada, M., Aoyagi, M.: Theory and Application of Learning System. Morikita, p. 195 (2005) (Japanese)
16. Nagata, K., Watanabe, S.: A proposal and effectiveness of the optimal approximation for Bayesian posterior distribution. In: Workshop on Information-Based Induction Sciences, pp. 99–104 (2005)
17. Takamatsu, S., Nakajima, S., Watanabe, S.: Generalization Error of Localized Bayes Estimation in Reduced Rank Regression. In: Workshop on Information-Based Induction Sciences, pp. 81–86 (2005)

A Study of Emotion Recognition and Its Applications

Won-Joong Yoon and Kyu-Sik Park

Dankook University
Division of Information and Computer Science
San 8, Hannam-Dong, Yongsan-Ku, Seoul Korea, 140-714
{helloril,kspark}@dankook.ac.kr

Abstract. In this paper, a speech emotion recognition system and its application for call-center system is proposed. In general, a speech captured by cellular-phone contains noises due to the mobile network and speaker environment. In order to minimize the effect of these noises and so improve the system performance, we employ a simple MA filter at the feature domain. Two pattern classification methods, k-NN and SVM with probability estimate, are compared to distinguish two emotional states- neutral and anger- for call-center application. The experimental results indicate that the proposed method provides very stable and successful emotional classification performance and it promises the feasibility of the agent for mobile communication services.

1 Introduction

Most of speech emotion classification has two stages of a pattern recognition problem: *feature extraction* and *classification* based on the selected feature. Depending on the various combinations of these stages, several strategies are employed in the literature. Dellaert et al. [1] used 17 features and compared three classifiers: maximum likelihood Bayes classification, kernel regression and k-NN (Nearest Neighbor). They reached 60% - 65% accuracy with four emotion categories. Scherer [2] extracted 16 features by the jack-knifing procedure and achieved an overall accuracy 40.4% for fourteen emotional states. As a first direct attempt for emotion recognition in call center application, G. Zhou et al. [3] used nonlinear Teager Energy Operator (TEO) feature for stressed/neutral classification and compared the feature performance with the traditional pitch and MFCC feature. Yacoub et. al. [4] focused on distinguishing anger versus neutral speech for call center applications and they achieved a maximum accuracy of 94%. However these papers did not report any analysis and results with an environmental and network noise effect in call center application. Other good works on general speech emotional classification can be found in [5-7].

In contrast to previous works, this paper focuses on the following issues of the emotion recognition problem.

Firstly, the proposed system accepts query sound captured by a cellular phone in real-time using INTEL Dialogic D/4PCI board. In order to minimize the effect of mobile noises and so improve the system performance, we adopt a simple MA (Moving Average) filter which has relatively simple structure and low computational complexity.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 455–462, 2007.

Secondly, a SFS (Sequential Forward Selection) feature optimization method is implemented to further improve and stabilize the system performance. Aim to our implementation we choose SVM [8] as our classification algorithm and compared the performance with simple k-NN algorithm.

Thirdly, for practical application, we considered call-center system that can recognize two emotional states – neutral and anger with a classification probability about each emotional state. Call-center system allows automatic call routing of negative customer and provide a feedback to an operator for monitoring purposes. This agent will be quite promising in various mobile applications.

This paper is organized as follows. Section 2 describes proposed mobile-based emotion recognition system. Section 3 explains the methods of noise-robust feature extraction and optimization. Section 4 compares experimental results of the proposed system and describes the applications. Finally, a conclusion is given in section 5.

2 Proposed Mobile-Based Emotion Recognition System

The emotion recognition engine consist 2 stages – speech signal acquisition, emotion pattern classification. A query speech signal is first picked up by the single microphone of the cellular phone and then transmitted to the emotion recognition server. The transmitted signal is acquired in real-time by using INTEL dialogic D4PCI-U board in 8 kHz sampling rate, 16 bit, MONO. Then the queried speech is classified with confidence probability (for the case of SVM) and the classification result will be reported and transmitted to the mobile agent.

Fig. 1 shows the block diagram of the proposed emotion classification engine.

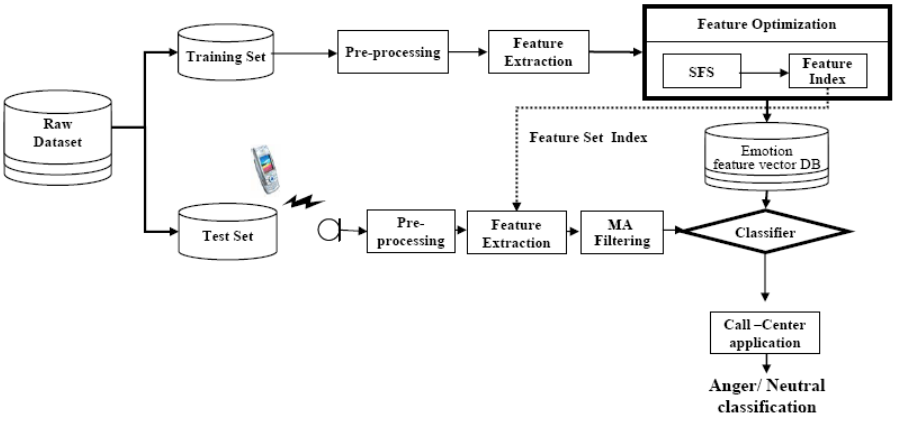


Fig. 1. Proposed emotion classification engine and the Call-center application

Raw speech data set is randomly divided into two sets of data set – training set and test set. At the training phase, feature set is extracted from pre-processed speeches in training set. Then a feature optimization is applied with respect to pattern classifier. These feature set is used to build up emotional feature vector DB.

On the other hand, in the testing phase, a test input speech is picked up by a cellular phone and pre-processed as before, and the same indexed feature set from the feature optimization stage is extracted. Then the MA filter is applied to these feature vectors to minimize the effect of background noise. Finally, the pattern classifiers are tested for queried speech and the performance of the system is evaluated for Call-center application.

3 Feature Extraction and Optimization

3.1 Pre-processing and Emotional Feature Extraction

Speech pre-processing stage is required to build up the robust emotional features as shown in fig. 2.

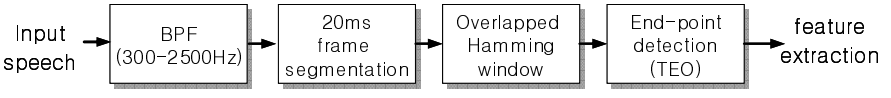


Fig. 2. Speech pre-processing procedure

The input speech signal is first band-pass filtered (BPF) with a passband 300Hz – 2500 Hz. The reason for this BPF is clear if we compare the spectrogram of the original clean speech and the speech acquired from the cellular-phone. Since the speech through the mobile network undergoes the two main distortions – bandwidth reduction to 300Hz -2500Hz and mobile noise, BPF allows extracting only the meaningful signal portion of the noisy speech.

At the sampling rate of 8 kHz, the filtered signal is divided into 20ms frames with 50% overlapped hamming window at the two adjacent frames. Then the end-point detection algorithm is applied to distinguish voiced and non-voiced segment from the input speech so that the emotional features are extracted only from the voiced portion of speech. In this paper, we used a Li. Gu's [9] end-point detection algorithm.

After the pre-processing, two types of emotional features are computed from every 20 ms frame- one is the prosodic features such as pitch and energy, and the other is speech phoneme feature such as mel-frequency cepstral coefficients (MFCC). The means and standard deviations of these original features and their delta values are computed over each frame to form a total of 32-dimensional feature vector. Followings are short descriptions of the features used in this paper.

For the pitch extraction, several algorithms are tested [10-12] and SHR[12] algorithm is used in this paper. SHR (Subharmonic-to-Harmonic Ratio) transforms the speech signal into the FFT domain, and then decides two candidate pitches by detecting the peak magnitude of the log spectrum. The final pitch is determined by comparing SHR to a certain threshold.

Short time energy of the speech signal provides a convenient representation that reflects amplitude variations.

MFCC is the most widely used feature in speech recognition. It captures short-term perceptual features of human auditory system. In this paper, 6^{th} order MFCC is used.

3.2 Feature Optimization

Not all the 32-dimensional features in the previous section are used for emotional state classification purpose. Some features are highly correlated among themselves and some feature dimension reduction can be achieved using the feature redundancy. In order to reduce the computational burden and so speed up the search process, while maintaining a system performance, an efficient feature dimension reduction and selection method is desired. In ref. [13], a sequential forward selection (SFS) method is used to meet these needs. In this paper, we adopt the same SFS method for feature selection to reduce dimensionality of the features and to enhance the classification accuracy. Firstly, the best single feature is selected and then one feature is added at a time which in combination with the previously selected features to maximize the classification accuracy]. This process continues until all 32 dimensional features are selected. After completing the process, we pick up the best feature lines that maximize the classification accuracy.

3.3 Noise Filtering Using MA Filter

A speech captured by cellular-phone contains noises due to the mobile network and speaker environment. In order to minimize the effect of these noises and so improve the system performance, we employ a simple MA(Moving Average) filter that has relatively simple structure and low computational complexity.

The key idea is to apply MA filter to the feature sequences on feature domain to smooth out spikes due to the mobile noises. In order to apply MA filter, we represent feature data by two dimensional matrix FM_{td}. Here, each row of FM_{td} represents a feature vector and each column represents a time sequence. Then the features in column order are normalized to zero mean and unit variance in order to avoid numerical problems caused by small variances of the feature values. This normalized time sequence of features is further processed by MA filter as follows.

$$\hat{x}_{td} = \frac{1}{M}\sum_{i=0}^{M} x_{(t-i),d}\ , \quad x_{(t-i),d} = \text{feature coefficients} \tag{1}$$

In this paper, the filter order M=5 was found to yield the best results for our experiments. This simple feature post-processing step has a good positive influence on minimizing noise effect. We note that MA(2) filter is also applied to the features when we setup the emotional DB as shown in figure 1 and the reason for this is to minimize the differences in feature coefficients between the query and DB due to the MA filtering operation.

4 Experimental Results

4.1 Experimental Setup

We used the database from the Yonsei University [14]. The original data set includes short utterances by fifteen semi-professional actor and actress and it consists of total

of 5400 utterances across 2 emotional states, i.e., neutral and anger in 8 kHz, 16 bit recording. Among them, 1000 utterances (500 utterances for each emotional state) were chosen as training set to define an optimal feature set by SFS optimization stage and to setup the emotional feature vector DB. Another 500 utterances (250 utterances for each emotional state) are used as the test data set for system evaluation.

Fig. 3 shows a block diagram of experimental setup. In order to compare the system performance, two sets of experiment have been performed.

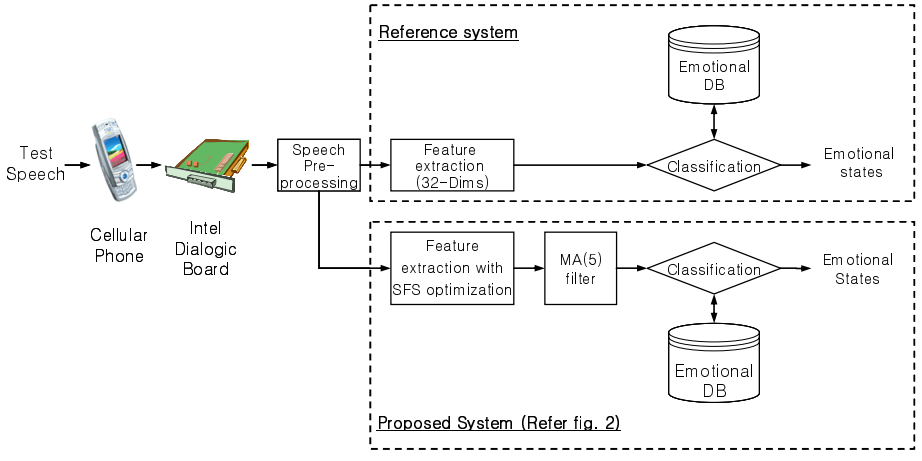


Fig. 3. Experimental setup for performance comparison of the proposed system

One is the system (called "proposed system") with a proposed method as in fig. 1 and the other is the system (called "reference system") without any noise reduction technique and the feature optimization method. Here the proposed system works as follows. Firstly, a test speech is acquired through actual cellular phone via INTEL dialogic D4PCI-U board in 8 kHz sampling rate, 16 bit, MONO. Secondly, emotional features are extracted according to the feature index from the SFS feature optimization stage and then MA(5) filter is applied to feature vector in order to minimize the effect of mobile noises. Finally the k-NN, SVM pattern classifier is tested for the classification of queried speech and the performance of the system is evaluated.

4.2 SFS Feature Optimization

Fig. 4 describes the SFS feature optimization procedure to select best feature coefficients with k-NN and SVM classifiers. As we see, performance increases with the increase of features up to certain number of features, while it remains almost constant after that. Thus we can select first 10 features up to the boundary and ignore the rest of them. For both k-NN/SVM classifiers, we observed that pitch and MFCC feature is most significant in the classification result. As we intuitively know, the less number of feature set is always desirable.

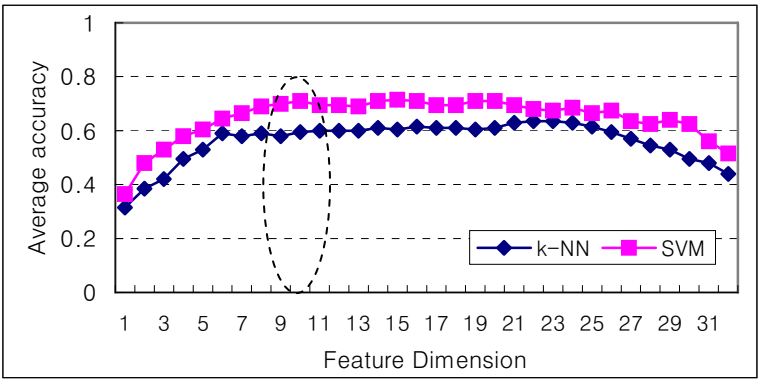


Fig. 4. Feature optimization procedure

4.3 Call-Center Application

For a practical application, emotion classification engine for call-center is implemented to distinguish two emotional states – neutral versus anger speech which is salient to call center application. This kind of system allows automatic call routing of negative customer and provide a feedback to an operator or a supervisor for monitoring purposes.

Table 1 compares the classification results between the proposed system and the reference system under the experimental setup as shown in fig. 3.

Table 1. Performance comparison of proposed system

	Reference System		Proposed System	
	k-NN (32-Dims)	SVM (32-Dims)	k-NN (18-Dims)	SVM (16-Dims)
Classification Accuracy	67.5%	72%	73.5%	86.5%

As seen from Table 1, we see much improvement of classification performance in proposed system over the reference system. It actually can achieves more than 8% ~ 14.5% improvement even with fewer number of feature set derived from SFS feature optimization. In terms of the classifier, SVM outperforms k-NN about 13%. In general, SVM is very well known to have low computation complexity, and so it is better idea to use SVM than k-NN classifier in terms of both classification accuracy and computational complexity.

Table 2 compares the classification performance between the proposed system and reference system in a form of a confusion matrix. Here (a) and (b) shows the classification result with k-NN and SVM classifier respectively for the reference system. On the other hand, (c) and (d) describes the results for the proposed system. The numbers

Table 2. Comparison of classification results in a form of confusion matrix. (a) reference system with k-NN, (b) reference system with SVM, (c) proposed system with k-NN, (d) proposed system with SVM

	Reference System				Proposed System			
	(a) k-NN		(b) SVM		(c) k-NN		(d) SVM	
Query	Neutral	Anger	Neutral	Anger	Neutral	Anger	Neutral	Anger
Neutral	73	27	46	54	68	32	80	20
Anger	38	62	2	98	21	79	7	93
Average Accuracy	67.5%		72%		73.5%		86.5%	

of correct classification result lies in the diagonal term and misclassification statistics are listed in off –diagonal terms.

As seen on Table 2, in reference system, k-NN classifier shows the misclassification rate 27% in anger state and 38% in neutral state, while SVM classifier shows the misclassification rate 54% in neutral state, but the anger state is easily recognizable with only 2% error. For the case of the proposed system, in k-NN classifier, the misclassification rate 32% for neutral state and 21% for anger state, so we see some improvements in classification of anger category. On the other hand for the SVM classifier, we see a significant improvement over the normal category where the misclassification rate decreases from 54% to 20%. From the Table 2, we observe that the neutral state is quite susceptible to the noises, while the anger state is quite easily recognizable even under the noise conditions.

5 Conclusion

In this paper, a speech emotion recognition system and its application for call-center system is proposed. The proposed system can recognize two emotional states - neutral and anger from the speech captured by a cellular-phone in real time. In order to alleviate noise effect due to the mobile network and speaker environment, we adopt a simple MA filter on the feature domain. SFS feature optimization is also utilized to improve and stabilize the system performance. The proposed system has been tested and compared with cellular phones in the real mobile environment and it shows a very stable and successful emotional classification performance and it promises the feasibility of the agents for mobile communication services. Future work will involve the development of new emotional features and further analysis of the system for practical implementation.

Acknowledgment

This work was supported by grant No. R01-2004-000-10122-0 from the Basic Research Program of the Korea Science & Engineering Foundation.

References

1. Dellaert, F., Polzin, T., Waibel, A.: Recognizing emotion in Speec. In: Proc. International Conf. on Spoken Language Processing, pp. 1970–1973 (1996)
2. Scherer, K.R.: Adding the affective dimension: A new look in speech analysis and synthesi. In: Proc. International Conf. on Spoken Language Processing, pp. 1808–1811 (1996)
3. Zhou, G., Hansen, J.H.L., Kaiser, J.F.: Nonlinear Feature Based Classification of Speech Under Stress. In: IEEE Transactions on speech and audio processing, vol. 9(3), IEEE Computer Society Press, Los Alamitos (2001)
4. Yacoub, S., Simske, S., Lin, X., Burns, J.: Recognition of emotions in interactive voice response system. In: Eurospeech 2003 Proc. (2003)
5. Kostov, V., Fukuda, S.: Emotion in user interface, Voice Interaction system. In: IEEE Intl. Conf. on systems, Man, Cybernetics Representation, vol. 2, pp. 798–803 (2000)
6. Oriyama, T.M., Oazwa,: Emotion recognition and synthesis system on speech. In: IEEE Intl. Conference on Multimedia Computing and Systems, pp. 840–844 (1999)
7. Lee, C.M., Narayanan, S., Pieraccini, R.: Classifying emotions in human-machine spoken dialogs. In: ICME'02 (2002)
8. Wu, T.-F., Lin, C.-J., Weng, R.C.: Probability Estimates for Multi-class Classification by Pairwise Coupling. Journal of Machine Learning Research (2004)
9. Gu, Li., Zahorian, S.A.: A new robust algorithm for isolated word end-point detection. In: ICASSP2002, Orlando, USA (2002)
10. Noll, M.: Pitch determination of human speech by the harmonic product spectrum, the harmonic sum spectrum, and a maximum likelihood estimate. In: Proceedings of the Symposium on Computer Processing Communications, pp. 779–797 (1969)
11. Ross, M.J., Shaer, H.L., Cohen, A., Freudberg, R., Manley, H.J.: Average magnitude difference function pitch extractor. ASSP-22, 353–362 (1974)
12. Sun, X.: A pitch determination algorithm based on subharmonic-to harmonic ratio. In: ICSLP, pp. 676–679 (2000)
13. Liu, M., Wan, C.: A study on content-based classification retrieval of audio database. In: Proc. of the International Database Engineering & Applications Symposium, pp. 339–345 (2001)
14. Kang, B.-S.: A text-independent emotion recognition algorithm using speech signal, MS Thesis, Yonsei University (2000)

Error Detection and Correction Based on Chinese Phonemic Alphabet in Chinese Text

Chuen-Min Huang, Mei-Chen Wu, and Ching-Che Chang

Department of Information Management
National Yunlin University of Science & Technology, Taiwan, R.O.C.
{huangcm,g9323734,g9523717}@yuntech.edu.tw

Abstract. Misspelling and misconception resulting from similar pronunciation appears frequently in Chinese texts. Without double check-up, this situation is getting even worse with the help of Chinese input method editor. It is hoped that the quality of Chinese writing would be enhanced if an effective automatic error detection and correction mechanism embedded in text editor. Therefore, the burden of manpower to proofread shall be released. Until recently, researches on automatic error detection and correction of Chinese text have undergone many challenges and suffered from bad performance compared with that of Western text editor. In view of the prominent phenomenon in Chinese writing problem, this study proposes a learning model based on Chinese phonemic alphabet. The experimental results demonstrate this model is effective in finding out most of words spelled incorrectly, and furthermore this model improves detection and correction rate.

Keywords: Error detection of Chinese text, Error correction of Chinese text, language model, Chinese phonemic alphabet.

1 Introduction

According to a survey about the employment preference of enterprises in 2005, it indicated that there were many misspellings in the résumés of applicants[20]. A misspelling refers to the correct word replaced by the word with similar shape or similar pronunciation. An earlier research indicates that misspellings of Chinese word mainly attribute to phonemic misuse. Furthermore, regardless of age, students usually will not accurately detect the correct word with similar or identical pronunciation.[18]. In recent years, the convenience of the Internet and the rapid improvement of Chinese character input methods have deteriorated the problem of misspelling caused by phonetics. In the past, the characters replaced by auxiliary words such as ei (ㄟ) (emphasis) or soundnification such as ker ker (ㄎㄎ) (laugh) do not make troubles for readers However, the pursuit of handiness and speed, using the word with similar or identical pronunciation as a substitute will not only perplex readers but also sometimes be perceived as ridiculous implication. For example, the phrase同聲勸'諫' (tung2 sheng1 chiuan4 jian4) is pronounced exactly the same as 同聲勸'賤', while the last word makes their meaning totally opposite. What is meant by

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 463–476, 2007.

the former phrase indicating the strong voices calling for righteous reformation; while the latter phrase means persuading someone for running dirty errands. For fun or for fame, the new created, novel words and phrases emerge at an unexpected speed in our daily life. In Japan, the publishing industry insists that it shall not allow misspelling existing in publication. To guarantee press quality, the industry endeavors to proofread each publication by human efforts. A report reveals that the general administration of press and publication of the People's Republic of China regulates the error rate of the publication to be kept within one in ten thousand [5]. While the issue of misspelling detection and correction raises public attention in developed countries, it is rare to have similar studies in pan-Chinese research communities. The earliest study related to this issue begins in 1990s focusing only on error detection [2]. The correction mechanism was proposed in later study, yet the result was not encouraging. There was no further study since then. This study aims to elaborate on both detection and correction in Chinese text processing. It is hoped that the proposed issue will raise more discussions in the near future.

2 Literature Review

2.1 Automatic Detection and Correction in Text

Research on automatic detection and correction of English text has focused on three difficult problems: (1) non-word error detection; (2) isolated-word error correction; and (3) context-dependent word correction. The two main techniques that have been explored for non-word error detection are n-gram analysis and dictionary lookup. N-gram error detection technique is conducted by examining the validity of each n-gram (bi-gram, tri-gram, …etc.) from the input string based on frequency consideration. The beauty of adopting this method is its dictionary independent and its ability to extract unknown words. Non-word error detection mainly adopts dictionary-based or n-gram segmention technique to examine the validity of the input string. Isolated-word error correction not only utilizes n-gram analysis and dictionary lookup but also adopts hybrid techniques such as minimum edit distance, similarity keys, rule-based procedures, probability estimates, and neural nets[17]. Context-dependent word correction utilizes Natural Language Processing (NLP) or statistical modeling. The typical NLP system consists of the procedures including parsing execution, lexical analysis, and semantic consideration.

There exists disparate procedures of error detection and correction process between Chinese and English languages due to their difference of morphology and syntax. Chinese language has no delimiters to separate words in a sentence. An earlier research suffering from low precision and no correction in text composing of word segmentation and scoring represents the connection strength of the singleton combined with the right-hand and left-hand neighboring characters[2]. Zhang et al.[21] constructed a confusing set considering the similarity of the code of Chinese Wu-Bi (five strokes) input method. Utilizing a confusing set and a classifier based on two-word linkage, POS-tag linkage, semantic feature and linkage in a word archives detection and correction of errors in Chinese text. To correct suspicious words, candidates are provided by the list of common wrongly written/pronounced words

with similar input code and concurrent characters in the context. As to the error types in Chinese, Zhang et al.[22] classified errors as non-word errors including character substitution error, string substitution error, character insertion error and character deletion error and real word containing word substitution error, word insertion error and word deletion error.

2.2 Unknown Word Detection

One of the most common problems in processing Chinese characters is the identification of valid words. Because there are no delimiters to separate words, the process of word identification encounters great difficulties in detecting unknown words. Unknown words include abbreviation, proper names, derived words, compounds and numeric type compounds. Among all, compounds and proper names are most widely known. If context contains error words, the process of unknown word detection becomes more complicate. It is because unknown word detection relies on contextual information. Until now, there is still no satisfactory algorithm for identifying both unknown words and typographical errors simultaneously. As the matter of fact, detection of unknown words and typographical errors share the same detection process. If all the known words in an input text can be detected, then the rest of the character string exists probable unknown words. According to an examination of a testing corpus, 4572 occurrences out of 4632 unknown words were incorrectly segmented into sequences of shorter words, and each sequence contained at least one monosyllabic word [3]. In other words, the appearance of monosyllabic word may come from the rest of unknown words. Therefore, the process of detecting unknown words is equivalent to making a distinction between monosyllabic words and monosyllabic morphemes which are parts of unknown words.

2.3 Language Model

Researches on NLP have lasted for a half century in order to overcome language gap between human and computer. During a half century, one of three achievements is the statistical language model applied in many application systems, such as speech recognition [10]. The classic task of language modeling aims to predict the probability of a sentence made up of a sequence of words. Suppose that there is a given sentence S made of a sequence of T words. They are $W_1, W_2, \ldots, W_T$. The probability of sentence S can be written as equation (1):

$$\begin{aligned} P(S) &= P(W_1, W_2, ..., W_T) \\ &= P(W_1)P(W_2 \mid W_1)...P(W_T \mid W_1 W_2 ... W_{T-1}) \\ &= \prod_{i=1}^{T} P(W_i \mid W_1 W_2 ... W_{i-1}) \end{aligned} \tag{1}$$

However, it is impossible to calculate the probability of all sequences of words due to the high complexity of calculation. Therefore, N-gram model based on Markov assumption that only the prior local context, the last few words, affects the next word to estimate the probability of a sequence of words reasonably[12, 13]. Consequently, the equation (1) is rewritten as equivalent equation (2):

$$P(S) = P(W_1, W_2, ..., W_T) \cong \prod_{i=1}^{T} P(W_1 \mid W_{i-n+1}^{i-1}) \tag{2}$$

In equation (2), W_{i-n+1}^{i-1} is $W_{i-(n-1)}W_{i-(n-2)} \ldots W_{i-1}$. The case of N-gram model that people usually use is for n=2, and this alternative is usually referred to as a bi-gram. N-gram model in general is to suggest using the relative frequency as a probability estimate. It denotes that the frequency of $W_{i-(n-1)}W_{i-(n-2)} \ldots W_{i-1}W_i$ in the training text is divided by the frequency of $W_{i-(n-1)}W_{i-(n-2)} \ldots W_{i-1}$ in the training text as in equation (3). $C(W_i^j)$ is the frequency of $W_{i-(n-1)}W_{i-(n-2)} \ldots W_{i-1}W_i$ in training text.

$$P(W_i \mid W_{i-n+1}^{i-1}) = \frac{C(W_{i-n+1}^{i})}{C(W_{i-n+1}^{i-1})} \tag{3}$$

2.3.1 Witten-Bell Smoothing

Smoothing techniques are used to resolve data sparseness of the language model. The popular Witten-Bell Smoothing technique is widely used to enhance N-gram model. The key concept of Witten-Bell Smoothing is to use a count of n-grams seen at least once to re-estimate the count of the unseen n-grams[19]. N-gram model is adjusted as follows:

$$P_{\text{int}\,erp}(W_i \mid W_{i-n+1}^{i-1}) = \lambda_{W_{i-n+1}^{i-1}} P(W_i \mid W_{i-n+1}^{I-1}) + (1 - \lambda_{W_{i-n+1}^{i-1}}) P_{\text{int}\,erp}(W_i \mid W_{i-n+2}^{i-1}) \tag{4}$$

Witten-Bell Smoothing is defined recursively as a linear interpolation of the maximum-likelihood estimate and the lower-order (n-1) gram model. $\lambda_{W_{i-n+1}^{i-1}}$ is a weight to incorporate n-gram with (n-1)-gram. This weight is calculated by equation (5).

$$1 - \lambda_{W_{i-n+1}^{i-1}} = \frac{P(W_i \mid W_{i-n+1}^{I-1}) + (1 - \lambda_{W_{i-n+1}^{i-1}}) P_{\text{int}\,erp}(W_i \mid W_{i-n+2}^{i-1})}{N_{1+}(W_{i-n+1}^{i-1}) + \sum_{W_i} C(W_{i-n+1}^{i})} \tag{5}$$

$N_{1+}W_{i-n+1}^{i-1}$ in equation (5) is the number of different words (types) that occur to the right of W_{i-n+1}^{i-1} :

$$N_{1+}W_{i-n+1}^{i-1} = \left|\{W_i : C(W_{i-n+1}^{i-1}W_i) > 0\}\right| \tag{6}$$

Witten-Bell Smoothing considers a number of word types preceding a word. If word types are rare, $\lambda_{W_{i-n+1}^{i-1}}$ shall increase. On the contrary, the probability of lower order n-gram is assigned more to the zero count n-grams. Because of the modifications of Witten-Bell Smoothing, the N-gram model is more appropriate.

3 System Architecture

The system architecture of this research is divided into three main parts including error detection, error correction and database as shown in Fig. 2. The database contains language model, lexicon and confusing word set. Error detection phase includes word segmentation of Chinese Knowledge and Information Processing (CKIP), and the process of dubious word area formation. Error correction phrase subsumes lexical analysis and optimal word extraction.

3.1 Error Detection

Error detection in Fig.2 is divided into word segmentation, and dubious word area formation. When the procedure of error detection starts, this system connects to word segmentation system of CKIP and initializes the service, then the process of unknown word extraction is activated. On completion of the process of unknown word extraction, the segmentation result will be obtained. The detailed processing and steps of detection and extraction will also be recorded. Moreover, the result of extracted unknown word detection will be manipulated further. First, the result of extracted unknown word detection is to be grouped into sentences. Second, each sentence will be segmented into words. Third, each tag of words will be filtered out. Fourth, words containing question mark will be highlighted and their location in text is to be notated.

3.1.1 Word Segmentations

The technique of word segmentation is adopted to execute unknown word extraction. It takes advantage of unknown word detection to detect misspelling of text.

3.1.1.1 Unknown Word Extraction. The word segmentation system of CKIP including 100,000-entry lexicon with pos tags, word frequencies, pos tag bi-gram information, etc. is to extract unknown word, segment text into words and notate pos tags. Moreover, the word segmentation process considers morphology analysis and language model for unknown words and try to represent morphology of all kinds of unknown words as context free grammar to improve the extraction of the unknown word without significant statistical characteristics. Its steps of process are as follows: (1) Initial segmentation. (2) Unknown word detection. (3) Chinese name extraction. (4) Foreign name extraction. (5) Compounds extraction. (6) Bottom-up merging algorithm. (7) Re-segmentation. After the unknown word extraction process, the result of unknown word extraction is produced not only the word segmentation and unknown word list, but also the detailed processing and steps of unknown word detection and extraction.

3.1.1.2 Extract Unknown Word Detection Result. This research adopts the unknown word detection as error detection. For different types of unknown words and typographical errors, they may share the same detection process. During the detection of error word, an input text undergoes the maximum matching algorithm of initial segmentation to distinguish monosyllabic words from monosyllabic morphemes by contextual information, so misspelling of text could be found out. After the process of word segmentation, this system extracts the unknown word detection result from unknown word extraction result for further manipulating.

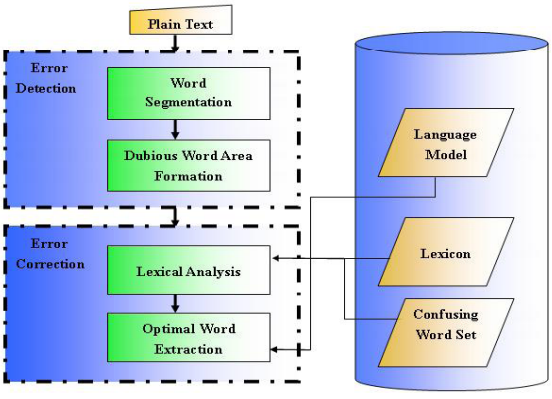


Fig. 1. System Architecture

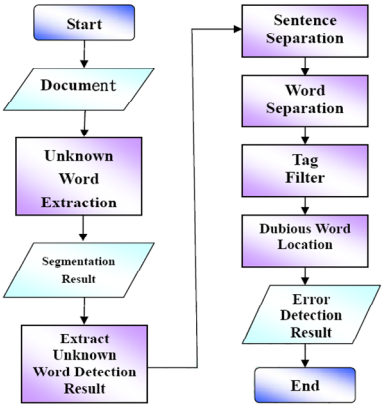


Fig. 2. Error Detection Procedures

3.1.2 Dubious Word Area Formation

Dubious Word Area Formation includes sentence separation, word separation, tags filter and dubious word location. After the process of unknown word detection, misspelling of input text could be found out. Furthermore, after the process of dubious word area formation, the location of misspelling could be exactly known, and then the misspelling could be manipulated.

3.1.2.1 Sentence Separation, Word Separation, Tags Filter and Dubious Word Location. A text containing dubious word would be separated to manipulate the sentence containing dubious word and present dubious word to users. During sentence separation, the location of each sentence of the text would be recognized and numbered. After sentence separation, the process of word separation is executed to separate and number words of the sentence containing dubious words and then to filter the tag of each word. After sentence separation, word separation and tags filter, the text in figure containing dubious words would be marked and presented to users.

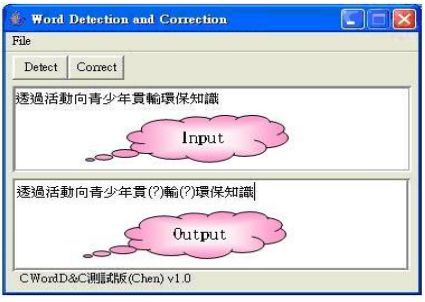


Fig. 3. Error Detection

3.2 Error Correction

The procedure of error correction is illustrated with Fig. 4. After the process of error diction, the location of dubious word could be known. Therefore, the dubious word and sentence could be easily extracted from text. Because the dubious word is probable error, the probable and correct words which are made of confusing word set and lexicon will compare with the context of sentence, and the maximum common strings will be extracted. And then the language model is used to select the optimal candidate word from the maximum common strings. The dubious word of sentence would be replaced by the optimal candidate word. The procedure of correction would be executed again and again until all dubious words are inspected.

3.2.1 Lexical Analysis

Lexical analysis includes the extraction of dubious sentence, word and candidate word, and word matching. After lexical analysis, the system would suggest candidate words.

3.2.1.1 Extract Dubious Sentence and Word. The process of error detection reveals not only whether dubious words exist in the text or not, but also the location of dubious sentences and words. When dubious words exist in the text, because the locations of dubious words are recorded, the process of error correction would easily extract the dubious sentence and word out. For example, the inputting sentence of Fig. 5 contains misspelling. The sentence of the text is the first sentence. Its location in the text is zero. The sentence is separated word by word. The misspelling in the sentence is the fifth word. Its location of the misspelling is four. According to its location, the character and word could be extracted from the detected document.

3.2.1.2 Extract Candidate Word. Because the dubious words are possible misspelling, candidate words are used to correct the dubious words. The candidate words are made up of confusing word set established according to the identical or similar phonetic properties, and lexicon provided by Institute of Information Science of Academia Sinica. For example, the candidate words in Fig. 6 are extracted from confusing word set and lexicon to be inspected.

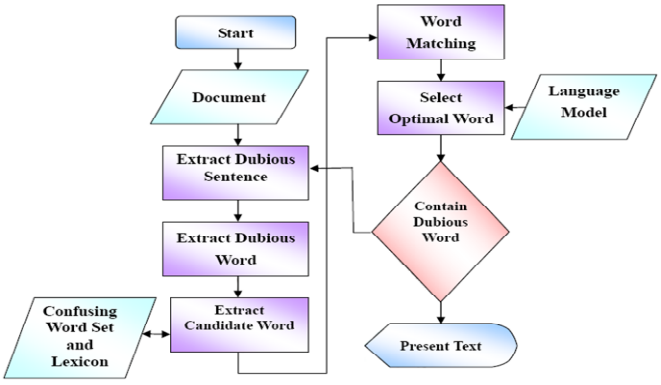


Fig. 4. Error Correction Procedure

Fig. 5. Extract Dubious Sentence and Word

3.2.1.3 Word Matching. Word matching is used to inspect whether the candidate words are suitable for the context of sentence. During the word matching, the candidate words would align the dubious words of the sentence and the comparing window would be reduced from the distance of the whole sentence to the longest distance over which the candidate words extent. And then the candidate words would compare with the words in the comparing window and the maximum common string would be extracted. For example, the sentence in Fig. 6 is in word matching. After word matching, because the longest word, Hai-Guan-Shu(海關署) does not match the words in the comparing window, it will be removed from the candidate words. The last suiting for the context of sentence would remain.

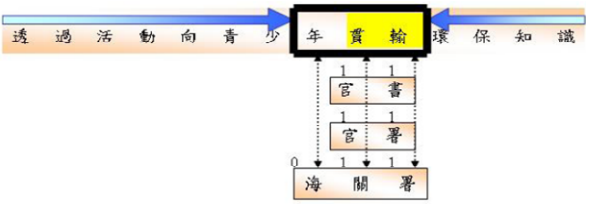


Fig. 6. Word Matching

3.2.2 Optimal Word Extraction

Optimal word extraction is used to select the most appropriate one of the candidate words existing in the sentence. The method for selecting the optimal word is language model. It is because we can obtain the possibility of connectivity between words from

language model to select the most possible word in the sentence. This research makes use of the character-based bi-gram language model and Witten-Bell Smoothing. The procedure to select optimal word is as follows:

1. The candidate words align the dubious words.
2. Adjust the windows size to the longest distance over which the candidate words extent plus one.
3. The candidate words replace the dubious word of the sentence.
4. Calculate the possibility of connectivity between words in the window with equation 7.
5. Choose the candidate word with the highest possibility as the optimal word.

After the procedure of optimal word extraction, the optimal word would be inserted into the sentence. Users could view not only the dubious word but also the optimal word. Take the sentence in Fig. 6 for example. After optimal word extraction, the optimal word is extracted by language model from the candidate words, as Fig. 7 illustrates that the optimal word is Guan-Shu (灌輸). After the detection and correction of misspelling, the result returns to users such as Fig. 8.

$$\log P(S) = \sum_{i=1}^{k} \log P_{\text{int}\,erp}(W_i \mid W_{i-n+1}^{i-1}) \tag{7}$$

3.3 Language Model

This Research adopts character-based bi-gram language model to gain the possibility of connectivity between words, and then the system could choose the optimal word because of the highest possibility of connectivity between words from language model. Furthermore, this research takes up the Witten-Bell Smoothing to resolve the data-sparseness problem. With the Witten-Bell Smoothing, the possibility of seen events could be reasonably distributed to the unseen events. Therefore the system could choose the optimal word more accurately.

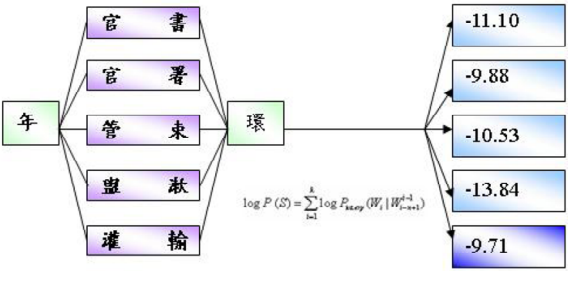


Fig. 7. Optimal Word Extraction

3.4 Confusing Word Set

The confusing word set is established according to the identical or similar phonetic properties. The task of confusing word set is to extract the probable correct word such as the pronounce of 灌 is similar to 貫.

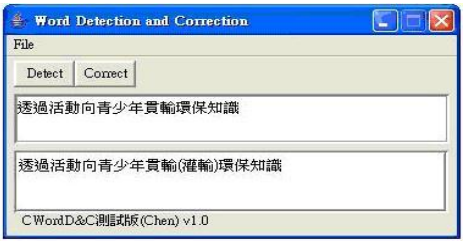


Fig. 8. Error Correction Example

3.5 Lexicon

The task of lexicon is to provide the candidate words. Because Lexicon contains correct words, the correct word could be provided from the lexicon to users such as 灌輸 (inculcate).

4 Experiment

4.1 Training Data Set

The task of training data set in the experiment of this research is to construct language model. In view of the convenience to acquire data and professional description of news documents, the news documents are the first choice to become the training data set. In order to own multi-descriptions of news documents, the sources of news documents would be from many of domestic and well-known online news websites. The online news websites are as follows. (1)United Daily News http://www.udn.com. (2)Central News Agency http://www.cna.com.tw/. (3)Broadcasting Corporation of China http://www.bcc.com.tw/. (4)China Times http://news.chinatimes.com/ (5)Formosa Television News http://www.ftvn.com.tw/. (6)TVBS News http://www.tvbs.com.tw/tvbs_page/index/. (7) Great News http://www.gnews.com.tw. (8)Reuters http://www.reuters.com. (9)Central Daily News http://www.cdn.com.tw/. (10)Taiwan daily news http:// www.taiwandaily.net/. Available news corpus is filled by all kinds of news topics such as politician, society, international, cross-straits, business, living, health, entertainment, leisure and technology. The news data downloaded from all the news websites by crawler that would be preserved as plain texts and then remove their tags. In order to ensure the quality of news data, there are five preprocessing to be checked:

1. Sum the occurrence counts of each single character word in all texts.
2. Inspect the rarely used words of news documents according to the frequency list provided by national languages committee of minister of education.
3. Inspect the particular formats.
4. Remove or behold the news data according to whether the occurrence of rarely used words or the particular formats is reasonable or not.
5. Select five hundred news documents at random to inspect their contents After completing the preprocessing, the fifty thousand news documents would be used to train language model. There are eighteen million, four hundred fifty thousand and seventy five words in news documents.

4.2 Confusing Word Set

Confusing word set is made of the word with identical or similar phonetic properties. Zhuyin Fuhao is the national phonetic system of the Republic of China for learning the Chinese languages. The system uses 37 special symbols to represent the Mandarin sounds: 21 consonants and 16 vowels. However, learning multi-languages influence the variations of pronunciation. For example, first, the retroflex sounds from Putonghua are softened considerably or the retroflex "r" ending is very rarely heard. Second, the dentilabial disappears. Third, the eng is pronounced as ong[1, 8]. It is the properties of the first variation of pronunciation that zh-, zhi becomes z-, zi; ch- becomes c-; sh-, shi becomes s-, si. The misconception of f- and hu- is the property of the second variation of pronunciation. The misconception of eng and ong is the property of the third variation of pronunciation. Therefore, besides three variation of pronunciation, this research includes the words with identical pronunciation or all tones to construct the confusing word set[15]. The steps of construction are as follows. (1) Construct the phonetic table of words. (2) Construct the homophone table. (3) Construct the confusing word set according to the five variation of pronunciation. After the completion of steps of construction, there are seven hundred and eighty-three collections with identical or similar pronunciation.

4.3 Lexicon

This research adopts lexicon provided by institute of information science of Academia Sinica to produce the candidate words because the lexicon contains correct words. The lexicon includes many types of words such as common noun, proper noun, idiom, parlance, derivative, variant, combinatorial word, jargon and dialects of Proto-Mandarin. Besides, there are words, their frequency and their pos-tag in the lexicon.

5 Experiment Results

The common misspellings from the books are used to evaluate the system[4, 6, 7, 9, 11, 14, 16]. And the common misspelling must be verified by Chinese dictionary. One hundred and thirty-seven documents with the common misspellings were retrieved from the Internet. There are two evaluations in the experiment. The first evaluation is to compare the results of detection and correction of Microsoft Word with those of this system. It is because the procedures of Microsoft Word divided apparently into the detection and correction of misspelling are identical to the system. The documents with misspellings would be highlighted and suggested by candidate words. Therefore, during the evaluation of system, documents with common misspelling would be respectively inputted into the system and Microsoft word to detect and correct the misspelling. The results of detection and correction from the two systems would be preserved and compared with each other. The second evaluation is to compare the result of the correction of Chinese natural input method with those of the system. For the reason that one task of Chinese natural input method provides the word automatically when inputting the sound of a word. So, the sentence with misspelling would be inputted into Chinese natural input method to check if it can select words correctly or not. At last the results of correction would compare with each other. The

first evaluation is to input the texts of documents with misspelling into Microsoft Word, activate the function of detection and correction, and then compare the results. After inspecting the result of detection and correction, one hundred and eight documents were detected accurately by Microsoft Word. Twenty-nine documents were not detected accurately. Thirty-eight documents were corrected accurately. Seventy documents were not corrected accurately. So, the detection rate is 0.788 and the correction is 0.352. The identical documents were inputted into this system. One hundred and eleven-one documents were detected accurately by this system. Twenty-six documents were not detected accurately. One hundred and three were corrected accurately. Eight documents were not corrected accurately. The evaluation of Microsoft Word and proposed model is as follows:

Table 1. Detection and Correction Rate

	Microsoft Word	Proposed Model
Detection Rate	108/137≅0.788	111/137≅0.810
Correction Rate	38/108≅0.352	103/111≅0.928

The second evaluation is to input the texts of documents with misspelling in Chinese natural input method, and compare the result. Eighty-nine documents were corrected accurately by Chinese natural input method. Forty-eight were not corrected accurately. The correction rate is 0.6496. One hundred and three documents were corrected by the system. Thirty-four documents were not corrected accurately. The correction rate is 0.7518. The evaluation of Chinese natural input method and proposed model is as follows:

Table 2. Correction Rate

	Chinese natural input method	Proposed Model
Correction Rate	89/137≅0.6496	103/137≅0.7518

The results of the two evaluations reveal that most of misspellings could be detected and corrected by this proposed model and the proposed model improves the detection and correction rate of the present models.

6 Conclusion and Future Work

This research focuses on the misspellings resulting from homophones and makes use of unknown word detection for error detection. It is because unknown word detection takes not only frequency of words into consideration but also morphemes of morphology. If only considering the frequency of words, the seldom used words will be neglected. Moreover, the characters appearing together frequently are not necessarily words. After the unknown word detection, the monosyllabic words and monosyllabic morphemes could be easily distinguished. This information could be

processed to recognize misspellings. The process of phonetic alphabets including homophone, tones to be undistinguished, and three characteristics resulting from learning multi-language is the foundation of error correction. The experiment result reveals that the proposed model could detect the misspellings and suggest the users the correct candidate words. So, users will easily examine their words in text to make sure whether there are misspellings in text or not. It is expected to release the local constrains of language model in our future work. Besides, it is a future undertaking for the adoption of a larger volume of balanced training data set to enhance the quality of language model and to meliorate the completion of morpheme.

References

1. Cao, F.F.: Instances of interaction between Taiwanese Japan and Taiwanese Mandarin in Taiwan across the span of the last one hundred year. Chinese Study 36(12), 273–297 (2000)
2. Chang, C.-H.: A New Approach for Automatic Chinese Spelling Correction. In: Proceedings of Natural Language Processing Pacific Rim Symposium ’95, Seoul, Korea, pp. 278–283 (1995)
3. Chen, K.J., Bai, M.H.: Unknown Word Detection for Chinese by a Corpus-based Learning Method. In: Computational Linguistics and Chinese Language Processing, pp. 27–44 (1998)
4. Chi, C.: You jian bie zi, 2nd edn. Ming Jen Publications, Inc. Taipei (1980)
5. Chiang, H.: tou shi:ti xiao jie fei di bu zhi shi cuo wu bai chu. In: Focus on China beijing: BBC CHINESE.com (2006)
6. Chuang, T.I., Chuang, S.Y.: Yi zi zhi cha. Jian Lin, Taipei (1991)
7. Fan, S.P.: Xiao yuan chang jian cuo bie zi shou ce. Chinese improvement working group, Hong Kong (1998)
8. Hsieh, K.P.: Ti wan di qu nian qing ren yu zh(ㄓ), ch(ㄔ), sh(ㄕ) with z(ㄗ), c(ㄘ), s(ㄙ) zhen di bu fen ma?Do young people in Taiwan really confuse zh(ㄓ), ch(ㄔ), sh(ㄕ) with z(ㄗ), c(ㄘ), s(ㄙ)?, The World of Chinese Language, vol. 90(12), pp. 1–7 (1998)
9. Hsieh, P.C.: Bie zai xie cuo zi liao. Business Weekly Publications, Inc. Taipei (2001)
10. Huang, C.N., Chang, H.F.: Zi ran yu yan chu li ji shu di san ge li cheng bei. Foreign Language Teaching and Research. 2005, 180–187 (2002)
11. Hung, F.L.: Bian zi ji jin. Fu Wen, Kaohsiung (1997)
12. Manning, C.D., Schütze, H.: Foundations of statistical natural language processing. MIT Press, Cambridge (1999)
13. Papoulis, A.: Probability, Random Variables, and Stochastic Processes, 2nd edn. McGraw-Hill, New York (1984)
14. Ssu Ma, T.: Cuo bie zi chu lie, 1st edn. Business Weekly Publications, Inc. Taipei (2005)
15. Ssu Tu, A.J.: Hao wan cuo bie zi you xi. Singtao, Hong Kong (2005)
16. Tso, H.L.: Cuo bie zi bian zheng. The Commercial Press, Ltd. Taipei (1980)
17. Wagner, R.A.: Order-n correction for regular languages. Commun. ACM 17, 265–268 (1974)
18. Wang, H.J.: Gao zhong zhi xue sheng zuo wen cuo bie zi yan jiu-yi gao xiong shi gao zhong zhi xue sheng zuo wen wei li. In: Wang, H.J. (ed.) Junior high school material. vol. Graduate Student, vol. 220, National Kaohsiung Normal University, Kaohsiung (2003)

19. Witten, I.H., Bell, T.C.: The zero-frequency problem: estimating the probabilities of novel events in adaptive text compression.Information Theory. IEEE Transactions 37, 1085 (1991)
20. Yang, H.I.: Xue shi zhong wen cheng du qi ye zhu guan yao tou. China times express, Taipei Report (2005)
21. Zhang, L., Zhou, M., Huang, C., Lu, M.: Approach in automatic detection and correction of errors in Chinese text based on feature and learning. In: Proceedings of the 3rd world congress on Intelligent Control and Automation, Hefei, pp. 2744–2748 (2000)
22. Zhang, L., Zhou, M., Huang, C., Pan, H.: Automatic detecting/correcting errors in Chinese text by an approximate word-matching algorithm. In: The 38th Annual Meeting of the Association for Computational Linguistics, Hong Kong (2000)

Mining Frequent Diamond Episodes from Event Sequences*

Takashi Katoh[1], Kouichi Hirata[2], and Masateru Harao[2,**]

[1] Graduate School of Computer Science and Systems Engineering
[2] Department of Artificial Intelligence
Kyushu Institute of Technology
Kawazu 680-4, Iizuka 820-8502, Japan
{f673024t,hirata,harao}@ai.kyutech.ac.jp
Tel: +81-948-29-7622, Fax: +81-948-29-7601

Abstract. In this paper, we introduce a *diamond episode* of the form $s_1 \mapsto E \mapsto s_2$, where s_1 and s_2 are events and E is a set of events. The diamond episode $s_1 \mapsto E \mapsto s_2$ means that every event of E follows an event s_1 and is followed by an event s_2. Then, by formulating the *support* of diamond episodes, in this paper, we design the algorithm FreqDmd to extract all of the *frequent diamond episodes* from a given event sequence. Finally, by applying the algorithm FreqDmd to bacterial culture data, we extract diamond episodes representing *replacement of bacteria*.

1 Introduction

It is one of the important tasks for data mining to discover frequent patterns from time-related data. Agrawal and Srikant [3,9] have introduced one method for such a task called *sequential pattern mining* to discover frequent *subsequences* as patterns in a sequential database. The sequential pattern mining has been developed by designing the efficient algorithms [8], in particular, with a non-redundant form called a frequent *closed* subsequence [12,13].

Mannila *et al.* [7] have introduced another method for such a task called *episode mining* to discover frequent *episodes* as patterns in an *event sequence*. By combining a *parallel episode* as a set of events and a *serial episode* as a sequence of events, the episodes are formulated as *directed acyclic graphs* of events of which edges specify the temporal precedent-subsequent relation. The episode mining has been developed by introducing the specific form of episodes for every target area together with efficient algorithms [4,5,6].

While the subsequence just represents linearly ordered relations as patterns, the episode can represent more complex relations. Also since the *frequency* of the episode is formulated by a *window* that is a subsequence of an event sequence under a fixed time span, the episode mining is more appropriate than the sequential pattern mining when considering the time span.

* This work is partially supported by Grand-in-Aid for Scientific Research 17200011 from the Ministry of Education, Culture, Sports, Science and Technology, Japan.
** Current address: Sojo University.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 477–488, 2007.

By focusing on the direct *causality* in an event sequence, Katoh *et al.* [5,6] have introduced a *sectorial episode* of the form $E \mapsto e$, where E is a parallel episode and e is an event. A sectorial episode is a direct precedent-subsequent relation between events. Then, they have designed an efficient algorithm Sect to extract all of the sectorial episodes that are frequent and accurate and apply the algorithm to bacterial culture data in order to extract sectorial episodes representing changes for drug resistance. However, it remains open to formulate an episode that represents an indirect precedent-subsequent relation beyond a sectorial episode, and then to design an algorithm to extract such episodes.

Hence, in this paper, we introduce a *diamond episode*, which is an extension of a sectorial episode, of the form $s_1 \mapsto E \mapsto s_2$, where s_1 and s_2 are events and E is a parallel episode. We call s_1 a *source* and s_2 a *sink*. The diamond episode $s_1 \mapsto E \mapsto s_2$ means that every event of E follows a source s_1 and is followed by a sink s_2, so we can regard every event in E as a *causation* through the change from a source s_1 to a sink s_2. Then, we formulate the *support* $supp(s_1 \mapsto E \mapsto s_2)$ of a diamond episode $s_1 \mapsto E \mapsto s_2$ as the ratio of the number of k-windows (i.e., a window with width k) in which $s_1 \mapsto E \mapsto s_2$ occurs for the number of all k-windows. For the *minimum support* σ such that $0 < \sigma < 1$, we say that $s_1 \mapsto E \mapsto s_2$ is *frequent* if $supp(s_1 \mapsto E \mapsto s_2) \geq \sigma$.

In this paper, we show that the diamond episode preserves *anti-monotonicity*, that is, for $E_1 \subseteq E_2$, if $s_1 \mapsto E_2 \mapsto s_2$ is frequent then so is $s_1 \mapsto E_1 \mapsto s_2$. Hence, under fixing a source s_1 and a sick s_2, we can deal with a diamond episode $s_1 \mapsto E \mapsto s_2$ as an itemset, by regarding a serial episode $s_1 \mapsto e \mapsto s_2$ with size 3 as an item for $e \in E$. Then, we design the algorithm FreqDmd to extract all of the *frequent diamond episodes* from a given event sequence by using the frequent itemset mining algorithm AprioriTid [1,2].

The algorithm FreqDmd first calls the algorithm BitSerl, which is a main difference from the algorithm Sect [5]. The algorithm BitSerl constructs the set of bit vectors representing the occurrences of all serial episodes with size 3, that is, of the form $s_1 \mapsto e \mapsto s_2$ for an event e, by using the bit-wise conjunction and disjunction, and the shift operator. Then, the algorithm FreqDmd extracts the set of all frequent diamond episodes of the form $s_1 \mapsto E \mapsto s_2$ for a parallel episode E, by regarding a serial episode $s_1 \mapsto e \mapsto s_2$ as an item e and by calling the algorithm AprioriTid [1,2].

Since our diamond episode is a combination of parallel and serial episodes, it is possible to extract all of the frequent diamond episodes by combining the algorithms designed in [7]. It is a main advantage for our algorithm FreqDmd to be efficient and appropriate for extracting frequent diamond episodes, because the algorithm FreqDmd scans a given event sequence just once and regards a serial episode as an itemset.

Finally, we apply the algorithm FreqDmd to bacterial culture data. Note that, from the medical viewpoint, in order to extract frequent diamond episodes concerned with *replacement of bacteria*, it is necessary to extract them based on the same sample. Hence, in this paper, by fixing the sample, we connect data of

every patient with some span, and then extract diamond episodes representing the replacement of bacteria from them.

2 Diamond Episode

As similar as [7], we assume that an event has an associated time of occurrence as a natural number. Formally, let $\mathcal{E}$ be a set of *event types*. Then, a pair (e, t) is called an *event*, where $e \in \mathcal{E}$ and t is a natural number which is the (*occurrence*) *time* of the event. In the following, for a set $E \subseteq \mathcal{E}$ of event types, we denote $\{(e, t) \mid e \in \mathcal{E}\}$ by (E, t), and also call it by an *event* again. Furthermore, we denote a set $\{e_1, \ldots, e_m\} \subseteq \mathcal{E}$ of event types by a string $e_1 \cdots e_m$.

An *event sequence* $\mathcal{S}$ on $\mathcal{E}$ is a triple (S, T_s, T_e), where

$$S = \langle (E_1, t_1), \ldots, (E_n, t_n) \rangle$$

is an ordered sequence of events satisfying the following conditions.

1. $E_i \subseteq \mathcal{E}$ $(1 \leq i \leq n)$,
2. $t_i < t_{i+1}$ $(1 \leq i \leq n-1)$, and
3. $T_s \leq t_i < T_e$ $(1 \leq i \leq n)$.

In particular, T_s and T_e are called the *starting* time and the *ending* time of $\mathcal{S}$. We denote $T_e - T_s$ by $l_{\mathcal{S}}$.

A *window* in an event sequence $\mathcal{S} = (S, T_s, T_e)$ is an event sequence $W = (w, t_s, t_e)$ such that $t_s < T_e$, $t_e > T_s$ and w consists of all of the events (e, t) in S where $t_s \leq t < t_e$. The time span $t_e - t_s$ is called the *width* of the window. We call a window with width k in $\mathcal{S}$ a *k-window*, and denote the k-window $(w, t, t + k)$ of $\mathcal{S}$ starting from t by $w(\mathcal{S}, t, k)$.

Note that we can regard a set $E = e_1 \cdots e_m$ of event types as a *parallel episode* and a sequence $e_1 \mapsto \cdots \mapsto e_m$ of event types as a *serial episode* [7]. In this paper, we newly introduce the following *diamond episode*, as an extension of a sectorial episode [5,6].

Definition 1. Let s_1 and s_2 be event types and $E \subseteq \mathcal{E}$ be a parallel episode. Then, a *diamond episode* is of the following form.

$$s_1 \mapsto E \mapsto s_2.$$

We call s_1 a *source* and s_2 a *sink*.

We call a parallel, a serial and a diamond episodes *episodes* simply.

Let $\mathcal{S}$ be an event sequence $\mathcal{S} = (S, T_s, T_e)$ and e an event type. Then, we say that e *occurs* in $\mathcal{S}$ if there exists an event $(E, t) \in S$ such that $e \in E$. We denote $\{t \mid (E, t) \in S \wedge e \in E\}$ by $T(e, \mathcal{S})$. Also we denote $st(e, \mathcal{S}) = \min\{t \mid t \in T(e, \mathcal{S})\}$ and $et(e, \mathcal{S}) = \max\{t \mid t \in T(e, \mathcal{S})\}$.

Definition 2. We say that a parallel episode E *occurs* in $\mathcal{S}$ if every $e \in E$ occurs in $\mathcal{S}$. Also we say that a serial episode $e_1 \mapsto \cdots \mapsto e_m$ *occurs* in $\mathcal{S}$ if every e_i occurs in $\mathcal{S}$ $(1 \leq i \leq m)$ and $st(e_i, \mathcal{S}) < et(e_{i+1}, \mathcal{S})$ $(1 \leq i \leq m-1)$.

We say that a diamond episode $s_1 \mapsto E \mapsto s_2$ *occurs* in $\mathcal{S}$ if s and t occur in $\mathcal{S}$, and, for every $e \in E$, e occurs in $\mathcal{S}$, $st(s_1, \mathcal{S}) < et(e, \mathcal{S})$ and $st(e, \mathcal{S}) < et(s_2, \mathcal{S})$.

Let $\mathcal{S}$ be an event sequence and k a natural number. Then, we denote the set of all k-windows by $W(\mathcal{S}, k)$. Also, for an episode X, we denote the set of all k-windows such that X occurs in $\mathcal{S}$ by $W(X, \mathcal{S}, k)$.

Note that we can number all k-windows in $W(\mathcal{S}, k)$ from $T_s - k$ to T_e. We call such a number i $(T_s - k < i < T_e)$ the *label* of the i-th k-window. For an event sequence $\mathcal{S}$ and an episode X, we identify $W(X, \mathcal{S}, k)$ with the set of all labels of k-windows in which X occurs in $\mathcal{S}$.

Example 1. Let $\mathcal{E} = \{a, b, c\}$. Then, Figure 1 describes an event sequence $\mathcal{S} = (S, 4, 10)$ on $\mathcal{E}$ where:

$$S = \langle (abc, 4), (ab, 5), (a, 6), (ab, 7), (abc, 8), (ab, 9) \rangle.$$

Also an event sequence $w = (\langle (ab, 5), (a, 6), (ab, 7), (abc, 8), (ab, 9) \rangle, 5, 9)$ is a 5-window of $\mathcal{S}$ starting from 5, that is, $w = w(\mathcal{S}, 5, 4)$.

For the above event sequence $\mathcal{S}$, it holds that $T(c, \mathcal{S}) = \{4, 8\}$, $st(c, \mathcal{S}) = 4$ and $et(c, \mathcal{S}) = 8$. Also there exist 10 5-windows, of which starting time is from 1 to 9. Furthermore, for the above window w, diamond episodes $a \mapsto ab \mapsto c$ and $b \mapsto abc \mapsto b$ occur in w, for example.

4	5	6	7	8	9
a	a	a	a	a	a
b	b		b	b	b
c				c	

Fig. 1. An event sequence S in Example 1

Let $\mathcal{S}$ be an event sequence, X an episode and k a natural number. Then, the *frequency* $freq_{\mathcal{S},k}(X)$ and the *support* $supp_{\mathcal{S},k}(X)$ of X in $\mathcal{S}$ w.r.t. k are defined as follows.

$$freq_{\mathcal{S},k}(X) = |W(X, \mathcal{S}, k)|,\ supp_{\mathcal{S},k}(X) = \frac{freq_{\mathcal{S},k}(X)}{|W(\mathcal{S}, k)|}.$$

Definition 3. Let σ be the *minimum support* such that $0 < \sigma < 1$. Then, we say that an episode X is *frequent* if $supp(X) \geq \sigma$.

Lemma 1 (Anti-monotonicity for diamond episodes). *Let E_1 and E_2 be parallel episodes such that $E_1 \subseteq E_2$, If $s_1 \mapsto E_2 \mapsto s_2$ is frequent, then so is $s_1 \mapsto E_1 \mapsto s_2$.*

Proof. It is sufficient to show that $W(s_1 \mapsto E_2 \mapsto s_2, \mathcal{S}, k) \subseteq W(s_1 \mapsto E_1 \mapsto s_2, \mathcal{S}, k)$. Suppose that $l \in W(s_1 \mapsto E_2 \mapsto s_2, \mathcal{S}, k)$ and let W_l be the l-th k-window in $\mathcal{S}$. Then, it holds that $W_l = w(W(s_1 \mapsto E_2 \mapsto s_2, \mathcal{S}, k), l, k)$. For every $e \in E_2$, it holds that $st(s_1, W_l) < et(e, W_l)$ and $st(e, W_l) < et(s_2, W_l)$. Since $E_1 \subseteq E_2$, it holds that $st(s_1, W_l) < et(e', W_l)$ and $st(e', W_l) < et(s_2, W_l)$ for every $e' \in E_1$, so $s_1 \mapsto E_1 \mapsto s_2$ occurs in W_l. Hence, it holds that $l \in W(s_1 \mapsto E_1 \mapsto s_2, \mathcal{S}, k)$. □

3 Algorithm to Extract Diamond Episode

In this section, we design the algorithm to extract all of the frequent diamond episodes from an event sequence $\mathcal{S}$, the minimum support σ and the width k of windows.

The algorithm BitSerl, described as Figure 2, computes a bit vector of the occurrences of all serial episodes with size 3, by scanning an event sequence. Here, for a bit vector $v \in \{0,1\}^*$, $|v|$ denotes the length of v, and v_i denotes the i-th bit of v for $1 \leq i \leq |v|$, that is, $v = v_1 \cdots v_{|v|}$. Then, $sft(v)$ is $v_2 \cdots v_{|v|}0$. For $v, w \in \{0,1\}^*$, $v \circ w$ denotes the concatenation of v and w. Furthermore, for $v, w \in \{0,1\}^*$ such that $|v| = |w|$, $v \wedge w$ and $v \vee w$ are bit-wise logical conjunction and bit-wise logical disjunction, respectively. In particular, $0(k)$ denotes the zero vector $0 \cdots 0$ with length k.

---

procedure BitSerl$(S, T_s, T_e, k, \mathcal{E})$
/* $\langle S, T_s, T_e \rangle$: event sequence, k: the width of windows, $\mathcal{E}$: event types */
/* initialization, where $B[e] = B[e][T_s] \cdots B[e][T_e - 1]$ */
for $t = T_s$ **to** $T_e - 1$ **do**
  foreach $e \in \mathcal{E}$ **do** $B[e][t] \leftarrow 0$;
/* transforming an event sequence to a set of bit vectors */
for $t = T_s$ **to** $T_e - 1$ **do**
  foreach $e \in E$ such that $(E, t) \in S$ **do** $B[e][t] \leftarrow 1$;
/* constructing a bit vector of a serial episode $a \mapsto b \mapsto c$ */
$l \leftarrow T_e - T_s + k - 1$; $T \leftarrow \emptyset$;
foreach $(a, b, c) \in \mathcal{E} \times \mathcal{E} \times \mathcal{E}$ **do begin** /* $|V[\cdot]| = |W[\cdot]| = l$ */
  $V[a] \leftarrow 0(k-1) \circ B[a]$; $V[b] \leftarrow 0(k-1) \circ B[b]$; $V[c] \leftarrow 0(k-1) \circ B[c]$;
  $W[c] \leftarrow V[c]$; $W[b \mapsto c] \leftarrow 0(l)$; $W[a \mapsto b \mapsto c] \leftarrow 0(l)$;
  for $d = 1$ **to** $k - 1$ **do begin**
    $W[a \mapsto b \mapsto c] \leftarrow sft(W[a \mapsto b \mapsto c]) \vee (V[a] \wedge sft(W[b \mapsto c]))$;
    $W[b \mapsto c] \leftarrow sft(W[b \mapsto c]) \vee (V[b] \wedge sft(W[c]))$;
    $W[c] \leftarrow sft(W[c]) \vee V[c]$;
  end /* for */
  $T \leftarrow T \cup \{W[a \mapsto b \mapsto c]\}$;
end /* foreach */
return T;

Fig. 2. The algorithm BitSerl

Lemma 2. *Let $\mathcal{S}$ be an event sequence $\langle S, T_s, T_e \rangle$. Then, a serial episode $e_1 \mapsto \cdots \mapsto e_n$ occurs in a window $w(\mathcal{S}, t, k)$ if and only if one of the following two statements holds.*

1. *A serial episode $e_1 \mapsto \cdots \mapsto e_n$ occurs in a window $w(\mathcal{S}, t+1, k-1)$.*
2. *An event e_1 occurs in a window $w(\mathcal{S}, t, 1)$ and a serial episode $e_2 \mapsto \cdots \mapsto e_n$ occurs in a window $w(\mathcal{S}, t+1, k-1)$.*

Proof. First, we show the if direction. If the statement 1 holds, then there exists a sequence $(a_1, t_1) \ldots, (a_n, t_n) \in S$ of events such that $e_i = a_i$ $(1 \leq i \leq n)$ and $t + 1 \leq t_1 < \cdots < t_n < t + k$. Since $t \leq t + 1$, the statement holds. On the other hand, if the statement 2 holds, then there exists a sequence $(a_1, t_1) \ldots, (a_n, t_n) \in S$ of events such that $e_i = a_i$ $(1 \leq i \leq n)$, $t \leq t_1 < t + 1$, and $t + 1 \leq t_2 < \cdots < t_n < t + k$. Since $t \leq t_1 < t + 1 \leq t_2 < \cdots < t_n < t + k$, the statement holds.

Next, we show the only-if direction by contraposition, that is, if both of the following statements hold, then a serial episode $e_1 \mapsto \cdots \mapsto e_n$ $(n \geq 2)$ does not occur in $w(\mathcal{S}, t, k)$.

1. (A) A serial episode $e_1 \mapsto \cdots \mapsto e_n$ does not occur in $w(\mathcal{S}, t + 1, k - 1)$.
2. (B) An event e_1 does not occur in $w(\mathcal{S}, t, 1)$ or (C) a serial episode $e_2 \mapsto \cdots \mapsto e_n$ does not occur in $w(\mathcal{S}, t + 1, k - 1)$.

Suppose that both of the above statements hold but a serial episode $e_1 \mapsto \cdots \mapsto e_n$ occurs in $w(\mathcal{S}, t, k)$. Then, there exists a sequence $(a_1, t_1), \ldots, (a_n, t_n) \in S$ of events such that $e_i = a_i$ $(1 \leq i \leq n)$ and $t \leq t_1 < \cdots < t_n < t + k$.

Suppose that the statement (B) holds. Then, it holds that either $t_1 < t$ or $t + 1 \leq t_1$. Since $t \leq t_1$, it holds that $t + 1 \leq t_1$. Hence, it holds that $t + 1 \leq t_1 < \cdots < t_n < t + k$, which contradicts the statement (A).

On the other hand, suppose that the statement (C) holds. If $t + 1 \leq t_2$, then it holds that $t + 1 \leq t_2 < \cdots < t_n < t + k$, which contradicts the statement (C), so it holds that $t_2 < t + 1$. Since $t \leq t_1$, and t_i is a natural number, it holds that $t_1 \leq t_2 < t + 1 \leq t_1 + 1$, that is, $t_1 = t_2$, which contradicts that $t_1 < t_2$.

Hence, a serial episode $e_1 \mapsto \cdots \mapsto e_n$ does not occur in $w(\mathcal{S}, t, k)$. □

Theorem 1. *The algorithm* BitSerl *is correct.*

Proof. For a window w and an episode X, let *occ* be a function that $occ(w, X) = 1$ if E occurs in w; 0 otherwise. Since $V[\cdot]$ and $W[\cdot]$ are bit vectors with length l, we denote $V[\cdot][i] = (V[\cdot])_i$ and $W[\cdot][i] = (W[\cdot])_i$ for $1 \leq i \leq l$. Then, the algorithm BitSerl initially sets $V[\cdot][i]$ and $W[\cdot][i]$ to the following forms:

$$\begin{aligned} V[e][i] &= occ(w(\mathcal{S}, i - (k - 1), 1), e) \ (e \in \mathcal{E}), \\ W[a \mapsto b \mapsto c][i] &= occ(w(\mathcal{S}, i - (k - 1), 1), a \mapsto b \mapsto c), \\ W[b \mapsto c][i] &= occ(w(\mathcal{S}, i - (k - 1), 1), b \mapsto c), \\ W[c][i] &= occ(w(\mathcal{S}, i - (k - 1), 1), c). \end{aligned}$$

Suppose that, before the d-th iteration of the for-loop, $W[\cdot][i]$ is the following forms:

$$\begin{aligned} W[a \mapsto b \mapsto c][i] &= occ(w(\mathcal{S}, i - (k - 1), d), a \mapsto b \mapsto c), \\ W[b \mapsto c][i] &= occ(w(\mathcal{S}, i - (k - 1), d), b \mapsto c), \\ W[c][i] &= occ(w(\mathcal{S}, i - (k - 1), d), c). \end{aligned}$$

By Lemma 2, it holds that $occ(w(\mathcal{S}, i-(k-1), d+1), a \mapsto b \mapsto c) = occ(w(\mathcal{S}, i-(k-1+1), d+1-1), a \mapsto b \mapsto c) \vee (occ(w(\mathcal{S}, i-(k-1), 1), a) \wedge occ(w(\mathcal{S}, i-(k-1+1), d+1-1), b \mapsto c)) = W[a \mapsto b \mapsto c][i+1] \vee (V[a][i] \wedge W[b \mapsto c][i+1])$. Similarly, it holds that $occ(w(\mathcal{S}, i - (k - 1), d + 1), b \mapsto c) = W[b \mapsto c][i + 1] \vee$

$(V[b][i] \wedge W[c][i+1])$ and $occ(w(\mathcal{S}, i-(k-1), d+1), c) = W[c][i+1] \vee V[c][i]$. Here, $W[\cdot][i+1]$ is corresponding to the shift operation *sft* for the i-th bit. Thus, after the d-th iteration of the for-loop, $W[\cdot][i]$ is the following forms:

$$W[a \mapsto b \mapsto c][i] = occ(w(\mathcal{S}, i-(k-1), d+1), a \mapsto b \mapsto c),$$
$$W[b \mapsto c][i] = occ(w(\mathcal{S}, i-(k-1), d+1), b \mapsto c),$$
$$W[c][i] = occ(w(\mathcal{S}, i-(k-1), d+1), c).$$

Hence, by repeating the for-loop k times, we can obtain the bit vector $W[a \mapsto b \mapsto c]$ representing the occurrences of a serial episode $a \mapsto b \mapsto c$. □

After constructing the set of bit vectors representing the occurrences of all serial episodes with size 3 by the algorithm BitSerl, we design the algorithm FreqDmd to extract all of the frequent diamond episodes under the minimum support σ as Figure 3.

---

procedure FreqDmd$(S, T_s, T_e, k, \mathcal{E}, \sigma)$ /* $\langle S, T_s, T_e \rangle$: event sequence */
/* k: the width of windows, $\mathcal{E}$: event types, σ: the minimum support */
$T \leftarrow$ BitSerl$(S, T_s, T_e, \mathcal{E}, k)$; $D \leftarrow \emptyset$;
foreach $(s_1, s_2) \in \mathcal{E}$ **do begin**
  $I \leftarrow \emptyset$;
  foreach $W[s_1 \mapsto e \mapsto s_2] \in T$ **do**
    $U[e] \leftarrow W[s_1 \mapsto e \mapsto s_2]$; $I \leftarrow I \cup \{U[e]\}$;
    /* e is regarded as an item */
  $F \leftarrow$ AprioriTid(I, σ);
  /* call AprioriTid [1,2], where F is the set of frequent itemsets */
  foreach $E \in F$ **do** $D \leftarrow D \cup \{s_1 \mapsto E \mapsto s_2\}$;
end /* foreach */
return D;

Fig. 3. The algorithm FreqDmd

The algorithm FreqDmd calls the frequent itemset mining algorithm AprioriTid, introduced by Agrawal and Srikant [2], because we can regard a serial episode $s_1 \mapsto e \mapsto s_2$ as an item under fixing a source s_1 and a sink s_2, by Lemma 1. Then, in the algorithm FreqDmd, a serial episode $s_1 \mapsto e \mapsto s_2$ is referred as an event e, and extracts the set of all frequent itemsets by using the set I of bit vectors of the occurrences of $s_1 \mapsto e \mapsto s_2$, which is corresponding to L_1 in the algorithm AprioriTid [2].

Theorem 2. *The algorithm* FreqDmd *extracts all of the frequent diamond episodes from an event sequence by scanning it just once.*

Proof. By Theorem 1, the algorithm BitSerl is correct. Then, by Lemma 1 and the correctness of AprioriTid, the algorithm FreqDmd is correct.

Furthermore, the algorithm FreqDmd scans an event sequence just once in the algorithm BitSerl. □

Example 2. Consider the event sequence $\mathcal{S}$ in Example 1 (Figure 1).

Figure 4 describes the set I of bit vectors in the algorithm FreqDmd obtained by applying the algorithm BitSerl to $\mathcal{S}$ under the window width is 5. Here, (s_1, s_2) is a pair of a source s_1 and a sink s_2 and D is a serial episode $s_1 \mapsto e \mapsto s_2$ corresponding to $U[e] = W[s_1 \mapsto e \mapsto s_2]$.

(s_1, s_2)	D	0	1	2	3	4	5	6	7	8	9
(a, a)	$a \mapsto a \mapsto a$	0	0	1	1	1	1	1	1	0	0
	$a \mapsto b \mapsto a$	0	0	1	1	1	1	1	1	0	0
	$a \mapsto c \mapsto a$	0	0	0	0	0	1	1	1	0	0
(b, a)	$b \mapsto a \mapsto a$	0	0	1	1	1	1	1	1	0	0
	$b \mapsto b \mapsto a$	0	0	1	1	1	1	1	1	0	0
	$b \mapsto c \mapsto a$	0	0	0	0	0	1	1	1	0	0
(c, a)	$c \mapsto a \mapsto a$	0	0	1	1	1	0	0	0	0	0
	$c \mapsto b \mapsto a$	0	0	1	1	1	0	0	0	0	0
	$c \mapsto c \mapsto a$	0	0	0	0	0	0	0	0	0	0
(a, b)	$a \mapsto a \mapsto b$	0	0	0	1	1	1	1	1	0	0
	$a \mapsto b \mapsto b$	0	0	0	1	1	1	1	1	0	0
	$a \mapsto c \mapsto b$	0	0	0	0	0	1	1	1	0	0
(b, b)	$b \mapsto a \mapsto b$	0	0	0	1	1	1	1	1	0	0
	$b \mapsto b \mapsto b$	0	0	0	1	1	1	1	1	0	0
	$b \mapsto c \mapsto b$	0	0	0	0	0	1	1	1	0	0

(s_1, s_2)	D	0	1	2	3	4	5	6	7	8	9
(c, b)	$c \mapsto a \mapsto b$	0	0	0	1	1	0	0	0	0	0
	$c \mapsto b \mapsto b$	0	0	0	1	1	0	0	0	0	0
	$c \mapsto c \mapsto b$	0	0	0	0	0	0	0	0	0	0
(a, c)	$a \mapsto a \mapsto c$	0	0	0	0	1	1	1	0	0	0
	$a \mapsto b \mapsto c$	0	0	0	0	1	1	1	0	0	0
	$a \mapsto c \mapsto c$	0	0	0	0	0	0	0	0	0	0
(b, c)	$b \mapsto a \mapsto c$	0	0	0	0	1	1	0	0	0	0
	$b \mapsto b \mapsto c$	0	0	0	0	1	1	0	0	0	0
	$b \mapsto c \mapsto c$	0	0	0	0	0	0	0	0	0	0
(c, c)	$c \mapsto a \mapsto c$	0	0	0	0	1	0	0	0	0	0
	$c \mapsto b \mapsto c$	0	0	0	0	1	0	0	0	0	0
	$c \mapsto c \mapsto c$	0	0	0	0	0	0	0	0	0	0

Fig. 4. The set I of bit vectors for a source s_1 and a sink s_2 in the algorithm FreqDmd. Here, D denotes a serial episode $s_1 \mapsto e \mapsto s_2$ corresponding to $U[e] = W[s_1 \mapsto e \mapsto s_2]$ in Example 2.

Then, Figure 5 describes all of the frequent diamond episodes from $\mathcal{S}$ under the minimum support 50%. Note that we can realize the algorithm AprioriTid [1,2] by repeating the bit-wise conjunction of bit vectors in the breadth-first manner while satisfying the minimum support.

(s_1, s_2)	F	D
(a, a)	$\{a, b, ab\}$	$a \mapsto a \mapsto a,\ a \mapsto b \mapsto a,\ a \mapsto ab \mapsto a$
(b, a)	$\{a, b, ab\}$	$b \mapsto a \mapsto a,\ b \mapsto b \mapsto a,\ b \mapsto ab \mapsto a$
(a, b)	$\{a, b, ab\}$	$a \mapsto a \mapsto b,\ a \mapsto b \mapsto b,\ a \mapsto ab \mapsto b$
(b, b)	$\{a, b, ab\}$	$b \mapsto a \mapsto b,\ b \mapsto b \mapsto b,\ b \mapsto ab \mapsto b$

Fig. 5. All of the extracted frequent diamond episodes from $\mathcal{S}$ under the minimum support 50% and the window width 5 in Example 2. Here, no frequent diamond episode is extracted when $(s_1, s_2) = (c, a)$, (c, b), (a, c), (b, c) and (c, c).

4 Empirical Results

In this section, by applying the algorithm FREQDMD to bacterial culture data of Osaka Prefectural General Medical Center from 1995 to 1998, which are complete data in [10], we extract frequent diamond episodes concerned with *replacement of bacteria*. Here, we regard a pair of "attribute=value" as an event type.

First, we fix the width of windows as 30 days. Since the database contains the patient information not related to replacement of bacteria such as date, gender, ward and engineer [10], it is necessary to focus the specified attributes. Then, for a diamond episode of the form $s_1 \mapsto E \mapsto s_2$, we select the attributes age, department, sample, fever, catheter, tracheo, intubation, drainage, WBC (white blood cell) count, medication, Urea-WBC, Urea-Nitocide, Urea-Occultblood, Urea-Protein, Urea-amount, the total amount of bacteria, and the sensitivity of antibiotics as the event types in E, and the detected bacterium as a source s_1 and a sink s_2, in whole 44 attributes.

From the medical viewpoint, in order to extract frequent diamond episodes concerned with replacement of bacteria, it is necessary to extract them based on the same sample. Hence, by fixing the sample, we connect data of every patient with the span of 30 days, which is the width of windows.

Then, Figure 6 describes the number of windows and all of the frequent episodes under the minimum support 0.2%. Here, the column "c-episodes" denotes the number of diamond episodes after selecting $s_1 \mapsto E_1 \mapsto s_2$ from the frequent diamond episodes that has no $s_1 \mapsto E_2 \mapsto s_2$ such that $E_2 \subset E_1$ and $supp(s_1 \mapsto E_1 \mapsto s_2) = supp(s_1 \mapsto E_2 \mapsto s_2)$. We call such a frequent diamond episode a frequent *closed* diamond episode, as similar as a *closed* pattern [11,12,13,14].

sample	windows	episodes	c-episodes
catheter/others	94597	1180562	7226
respiratory organs	170633	25629974	163476

Fig. 6. The number of windows and all of the frequent and the frequent closed diamond episodes under the minimum support 0.2%

Figure 7 describes the bacteria of a source and a sick in the frequent closed diamond episodes. Note that if the sample is respiratory organs, then the bacterium "Staphylococcus aureus" occurs at either a source or a sink in the frequent closed diamond episode under the minimum support 0.2%.

Figure 8 and 9 describe the sensitivity of antibiotics (Ant) that is resistant (R), intermediate (I) and susceptibility, for the samples of catheter/others and respiratory organs, respectively, such that a source and a sick are different in Figure 7. Here, antibiotics are benzilpenicillin (PcB), synthetic penicillins (PcS), augmentin (Aug), anti-pseudomonas penicillin (PcAP), 1st generation cephems (Cep1), 2nd generation cephems (Cep2), 3rd generation cephems (Cep3), anti-pseudomonas cephems (CepAP), aminoglycosides (AG), macrolides (ML), tetracyclines (TC), carbapenems (CBP), vancomycin (VCM) and RFP/FOM (RFPFOM).

sample	bacterium of a source	bacterium of a sick	c-episodes
catheter/others	Staphylococcus aureus	Staphylococcus aureus	6396
	Bacteroides fragilis	Bacteroides fragilis	596
	Pseudomonas aeruginosa	Pseudomonas aeruginosa	120
	Escherichia coli	Escherichia coli	18
	Enterococcus faecalis	Enterococcus faecalis	1
	Staphylococcus aureus	Pseudomonas aeruginosa	45
	Bacteroides fragilis	Enterococcus faecalis	22
	Enterococcus faecalis	Pseudomonas aeruginosa	22
	Escherichia coli	Staphylococcus aureus	4
	Bacteroides fragilis	Staphylococcus aureus	2
respiratory organs	Staphylococcus aureus	Staphylococcus aureus	140681
	Pseudomonas aeruginosa	Pseudomonas aeruginosa	10512
	Staphylococcus aureus	Pseudomonas aeruginosa	6586
	Pseudomonas aeruginosa	Staphylococcus aureus	5677
	Staphylococcus aureus	Enterobacter cloacae	10
	Staphylococcus aureus	Serratia marcescens	6
	Enterobacter cloacae	Staphylococcus aureus	3
	Staphylococcus aureus	Klebsiella pneumoniae	1

Fig. 7. The bacteria in a source and a sink in the frequent closed diamond episodes

bacterium of a source	bacterium of a sick		sensitivity
Staphylococcus aureus	Pseudomonas aeruginosa	45	PcS=R(2), PcAP=S(1), Cep1=R(8), AG=S(6), TC=R(1), CBP=S(14)
Bacteroides fragilis	Enterococcus faecalis	22	PcB=R(2), PcB=S(1), PcAP=S(16), Cep1=R(6), Cep3=S(4), AG=S(2), ML=S(22), TC=S(22), CBP=S(22), VCM=S(1)
Enterococcus faecalis	Pseudomonas aeruginosa	22	PcB=R(1), Cep1=R(3), Cep2=S(1), TC=S(7), CBP=S(11)
Escherichia coli	Staphylococcus aureus	4	Cep3=S(2), TC=S(4), CBP=S(3)
Bacteroides fragilis	Staphylococcus aureus	2	PcB=R(1)

Fig. 8. The sensitivity of antibiotics for the sample of catheter/others in Figure 7

Hence, we can observe that, if a pair of a source and a sink is different, then so is sensitivity of antibiotics. In particular, for the case of a source "Bacteroides fragilis" and a sick "Enterococcus faecalis" in Figure 8, all episodes contain the sensitivity that ML=S, TC=S and CBP=S. On the other hand, for the case of a source "Staphylococcus aureus" and a sick "Pseudomonas aeruginosa" and the case of its alternation in Figure 9, episodes contain the sensitivity of both R and S for antibiotics AG, TC and CBP, and the episodes for the former case contain ML=S but the episodes for the latter case do not.

bacterium of a source	bacterium of a sick		sensitivity
Staphylococcus aureus	Pseudomonas aeruginosa	6586	PcB=R(974), PcS=R(4390), Aug=R(964), PcAP=S(1508), Cep1=R(3868), Cep2=R(567), Cep3=S(289), CepAP=S(1236), AG=R(139), AG=S(1868), TC=R(1265), TC=S(111), CBP=R(95), CBP=S(1994), VCM=S(1968), RFPFOM=S(109)
Pseudomonas aeruginosa	Staphylococcus aureus	5677	PcB=R(635), PcS=R(2639), Aug=R(1118), PcAP=S(1334), Cep1=R(2726), Cep2=R(382), Cep3=S(270), CepAP=S(864), AG=R(120), AG=S(1566), ML=S(63), TC=R(1212), TC=S(88), CBP=R(6), CBP=S(1533),VCM=S(457), RFPFOM=S(45)
Staphylococcus aureus	Enterobacter cloacae	10	PcS=R(2), Cep1=R(1), AG=S(1), CBP=S(4)
Staphylococcus aureus	Serratia marcescens	6	AG=S(1), CBP=S(1)
Enterobacter cloacae	Staphylococcus aureus	3	PcS=R(1)
Staphylococcus aureus	Klebsiella pneumoniae	1	—

Fig. 9. The sensitivity of antibiotics for the sample of respiratory organs in Figure 7

We will report elsewhere the analysis of the extracted diamond episodes from the medical viewpoints deeply.

5 Conclusion

In this paper, we have introduced a diamond episode of the form $s_1 \mapsto E \mapsto s_2$, where s_1 and s_2 are events and E is a set of events. Then, by formulating the *support* of diamond episodes, we have shown that the diamond episode preserves anti-monotonicity. Also we have designed the algorithm BitSerl to construct the set of bit vectors representing the occurrences of all serial episodes with size 3, by using the bit-wise conjunction and disjunction, and the shift operator. Then, we have designed the algorithm FreqDmd to extract all of the *frequent diamond episodes* from a given event sequence. Finally, we have applied the algorithm FreqDmd to bacterial culture data in order to extract frequent diamond episodes representing replacement of bacteria.

It is main advantage for the algorithm FreqDmd to scan a given event sequence just once, as same as the algorithm AprioriTid [1,2] or Sect [5]. However, the algorithm FreqDmd is insufficient to give empirical results efficiently, so it is a future work to improve the algorithm FreqDmd more efficiently.

In the evaluation in Section 4, we have introduced a *closed* diamond episode, because of reducing the number of extracted episodes. On the other hand, there are many algorithm to extract frequent closed patterns such as LCM [11] and CHARM [14] for frequent itemsets and BIDE [12] and CLOSPAN [13] for frequent sequences. Then, it is a future work to design an algorithm to extract frequent closed diamond episodes directly, by combining LCM or CHARM instead of APRIORITID in the algorithm FREQDMD.

In this paper, we have given empirical results for bacterial culture data. It is an important future work to apply our algorithm FREQDMD to various data in order to extract frequent diamond episodes.

References

1. Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A.I.: Fast discovery of association rules. In: Fayyed, U.M., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R. (eds.) Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press, pp. 307–328 (1996)
2. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proc. 20th VLDB, pp. 487–499 (1994)
3. Agrawal, R., Srikan, R.: Mining sequential patterns. In: Proc. 11th ICDE, pp. 3–14 (1995)
4. Bettini, C., Wang, S., Jajodia, S., Lin, J.-L.: Discovering frequent event patterns with multiple granularities in time sequences. IEEE Trans. Knowledge and Data Engineering 10, 222–237 (1998)
5. Katoh, T., Hirata, K., Harao, M.: Mining sectorial episodes from event sequences. In: Todorovski, L., Lavrač, N., Jantke, K.P. (eds.) DS 2006. LNCS (LNAI), vol. 4265, pp. 137–145. Springer, Heidelberg (2006)
6. Katoh, T., Hirata, K., Harao, M., Yokoyama, S., Matsuoka, K.: Extraction of sectorial episodes representing changes for drug resistant and replacements of bacteria. In: Proc. CME'07, 2007 (to appear)
7. Mannila, H., Toivonen, H., Verkamo, A.I.: Discovery of frequent episodes in event sequences. Data Mining and Knowledge Discovery 1, 259–289 (1997)
8. Pei, J., Han, J., Mortazavi-Asi, B., Wang, J., Pinto, H., Chen, Q., Daya, U., Hsu, M.-C.: Mining sequential patterns by pattern-growth: The PrefixSpan approach. IEEE Trans. Knowledge and Data Engineering 16, 1–17 (2004)
9. Srikant, R., Agrawal, R.: Mining sequential patterns: Generalizations and performance improvements. In: Apers, P.M.G., Bouzeghoub, M., Gardarin, G. (eds.) EDBT 1996. LNCS, vol. 1057, pp. 3–17. Springer, Heidelberg (1996)
10. Tsumoto, S.: Guide to the bacteriological examination data set. In: Suzuki, E. (ed.)Proc. International Workshop of KDD Challenge on Real-World Data (KDD Challenge 2000), pp. 8–12 (2000)
11. Uno, T., Asai, T., Uchida, Y., Arimura, H.: An efficient algorithm for enumerating closed patterns in transaction databases. In: Suzuki, E., Arikawa, S. (eds.) DS 2004. LNCS (LNAI), vol. 3245, pp. 16–31. Springer, Heidelberg (2004)
12. Wang, J., Han, J.: BIDE: Efficient mining of frequent closed sequences. In: Proc. 20th ICDE (2004)
13. Yan, X., Han, J., Afshar, R.: CloSpan: Mining closed sequential patterns in large datasets. In: Proc. 3rd SDM (2003)
14. Zaki, M.J., Hsiao, C.-J.: CHARM: An efficient algorithm for closed itemset mining. In: Proc. 2nd SDM, pp. 457–478 (2002)

Modeling Decisions for the Time-Dependent Optimal Lane Configuration Policy with Queueing Theory and Mathematical Programming

Seongmoon Kim

Yonsei University, Shinchon-dong 134 Seodaemun-gu, Seoul, Korea
kimsm@yonsei.ac.kr

Abstract. We provide mathematical models to operate a toll plaza with the time-dependent lane configuration policy. To formulate toll operations in the problem we use the queueing theory and mathematical programming. The queueing theory is utilized to obtain the stability condition which requires the mean arrival rate less than the mean service rate in each lane and compute the mean waiting time in the queue. The mathematical programming is used to determine the time-dependent lane configuration to minimize the total waiting and operation costs. In order to apply the introduced mathematical models in real world problem, we provide a case study based on the actual traffic data collected and show how the time-dependent lane configuration policy can be achieved in each time period. By numerical evaluation we demonstrate the electronic toll collection (ETC) is an intelligent transportation system which achieves high throughput and maintains almost no wait time.

1 Introduction

Ever increasing traffic volume and corresponding congestion necessitate efficient design and management of toll plaza operations. Constructing a toll plaza with right capacity and then time-dependent optimal lane configuration against the non-stationary and stochastic traffic volume can help prevent long waits during the peak times at the toll plaza. Queueing theory, which is considered to best describe the traffic situations at the toll plaza, has rarely been applied to toll plaza operations. Some previous studies use $M/M/1$ queueing systems to model the toll plaza problem, but the assumption on the exponential service time is unrealistic in practice. Several papers have presented case studies or simulations without general mathematical models to find *good* sub-optimal solutions for specific locations as illustrated in Section 2. This paper is differentiated from the previous work by providing an $M/G/1$ queueing model, which is more realistic and mathematically tractable, and it is then integrated with mathematical programming models to determine the time-dependent *optimal* lane configurations policy in each time period at the toll plaza.

V. Torra, Y. Narukawa, and Y. Yoshida (Eds.): MDAI 2007, LNAI 4617, pp. 489–499, 2007.

The rest of the paper is organized as follows. Section 2 provides related literature. Determination of the lower bounds for the number of lanes to open in each payment type, the corresponding total capacity at the toll plaza, and measures of performance based on the $M/G/1$ queueing process are given in Section 3. The queueing and mathematical programming models are provided to obtain time-dependent lane configuration policy minimizing total waiting and operation costs. Next, a case study is presented in Section 4 to show application of the introduced methodologies in determining the time-dependent optimal lane configuration policy based on the real traffic data collected. Lastly, concluding remarks and future research directions are given in Section 5.

2 Literature Review

The optimization problem for the toll plaza operations are related to the queueing studies because arrival and service processes are basically not deterministic but stochastic in nature. Initial work on optimal control policies for $M/G/1$ queueing systems is in [1] and its extensions with balking and reneging are presented in [2] and [3]. Queueing systems with two servers are analyzed in [4]. General cases with more than two server queueing systems are examined in [5]. More extant literature on queueing studies is reviewed in [6].

Toll plaza operation is closely related to pricing issues. Controlling queue length through the use of pricing strategy is presented in [7]. Public utility pricing with uncertainty and congestion is investigated in [8]. Peak-load pricing and capacity planning under the stochastic demand and supply are presented in [9]. Peak-period congestion pricing with elastic demand is considered in [10].

Recently, there has been growing interest in utilizing ETC systems. The potential advantages of ETC system through diverse applications of experimental implementation is ascertained in [11]. The impact of ETC systems is quantitatively investigated through the simulation study in [12]. Benefits in traffic operations at a toll plaza with ETC systems are evaluated via a case study in Florida, USA in [13]. A traffic operation model at ETC systems based on deterministic assumptions is presented in [14]. $M/M/1$ queueing systems to compute the upper bounds of mean queue length and mean wait time are used in [6]. In this study a capacity planning model subject to a reliability constraint is developed and, subsequently, a work force model is presented. Benefit-cost evaluation of the ETC system is presented in [15]. The best combination of ETC lanes and toll discount to maximize welfare is examined in [16] .

This paper has been motivated by the fact that most previous studies do not use the general service time distribution in the queueing model due to mathematical complexity although it is more realistic in practice than the exponential distribution. Another motivation is that there are several simulation and case studies for particular locations, but a few provide general optimization models in terms of the mathematical programming, and none of them has integrated an $M/G/1$ queueing process in the optimization model for the toll operation problem. We first begin an $M/G/1$ queueing model in the next section.

3 Formulation

Queueing theory has been often used to model the vehicle traffics at the toll plaza. In this section we first present main components of the queueing model for toll operations. The main elements of the queueing model are arrival and service processes, number of servers, queue discipline, etc.

There are many types of toll lanes in different cities, states, and countries. Lanes at the toll plaza may be divided as follows: dedicated manual toll lanes, dedicated automatic coin machine lanes (hereafter referred to as ACM lanes, dedicated electronic toll collection (ETC) lanes with automatic vehicle identification technology, and any mix of the above mentioned dedicated lane types, such as mixed ACM and ETC lanes, mixed ACM and manual lanes, and mixed ETC and manual lanes [17].

As a representation of the real toll plaza system, a queueing model should be formulated by making some assumptions on the probability distribution which should be both sufficiently realistic and mathematically tractable. It is assumed in this study that arrivals exhibit Poisson processes. Statistical test shows that arrivals at the toll plaza follow the standard assumption of a Poisson process with exponential interarrival times. This Poisson assumption is adopted many times in the literature [18], [19].

Let ν be the mean total arrival rate at the toll plaza, and ν_i mean arrival rate for lane type i, $i = 1, 2, \cdots, T$. If we let θ_i be the mean proportion of motorists using lane type i, we have the relationship $\nu_i = \nu\theta_i$ for each i. Without losing reality, we may be able to assume that arrival percentage per lane type is almost uniformly distributed over all lanes. Therefore, the mean arrival rate per lane is determined by dividing the mean arrival rate per lane type ν_i by the number of lanes utilized for that type of collection χ_i, and we have ν_i/χ_i. To make the problem mathematically tractable, we assume arrivals are distributed this way [6].

Service time from the service start to completion does not include wait time of the motorist in the queue. In other words, it means pure transaction time paying the toll, excluding waiting time. Thus, the service time distribution is not affected whether the congestion level is high or low at the toll plaza, and the probability distribution of the service time is time-invariant between the peak and the off-peak hours. If the service time has an exponential distribution, the mathematics of the queueing model are most manageable. However, it is not realistic and the probability distribution of the service time at the toll plaza does not follow the nice exponential distribution since an exponential service time distribution has a much larger variation than the one in the real situation. Hence, instead of assuming the exponentially distributed service time, we assume that service times have general distributions with the mean service time, $1/\eta_i$, and the variance of service time, τ_i^2, for lane type i.

By assuming the Poisson arrival process where the mean arrival rate per lane type is uniformly distributed across each lane and the general probability distribution for the service time with the mean and the variance, we can formulate an $M/G/1$ queueing model. We first determine the capacity Ω to design the toll

plaza. Ω should be big enough to handle all motorists at any time, but its upper bound may be limited because of the geographical restrictions. The mean arrival rate needs to be smaller than the mean service rate for each lane type in order for the queueing system to be stable, that is, $\nu_i/\eta_i < \chi_i$ for all i. Otherwise, the queue will build up without bound. Let ξ_i denote the minimum value of χ_i satisfying the inequality $\nu_i/\eta_i < \chi_i$, and ζ_i the upper bound for the number of type i lanes to open. Therefore, the capacity Ω of the toll plaza should satisfy the relationship $\sum_i \xi_i \leq \Omega$ to efficiently process all vehicles.

If the mean arrival rates, the mean service times and the variance of the service time are given, the mean waiting times in the queue can be computed for the $M/G/1$ queueing process by using the $Pollaczek-Khintchine\ formula$ [20]. Γ_i, the mean waiting time of the vehicle in the queue for lane type i and, Γ, the total waiting time of all drivers at the toll plaza can be obtained as follows.

$$\Gamma = \sum_i \nu_i \Gamma_i = \sum_i \nu_i \left(\frac{\nu_i(\eta_i^2 \tau_i^2 + 1)}{2\eta_i(\chi_i \eta_i - \nu_i)} \right) \tag{1}$$

Next we should concern optimal operations of the toll plaza. The objectives of operating a toll plaza may be different depending on the time of a day and the traffic situations. The objective may be minimizing the total waiting time at the toll plaza during the rush hours by using full capacity because of the high traffic volume. On the other hand, during the off-peak times the traffic volume may be low and service standard might be well satisfied even with partial capacity operations.

To find the optimal lane configuration at the toll plaza $(\chi_1^*, \chi_2^*, \cdots, \chi_T^*)$ during the rush hours given several constraints, we develop a following mathematical programming model minimizing the total wait costs, C.

1) *Objective Function*: Let α denote the waiting cost per unit time. Then, the total waiting cost of all motorists at the toll plaza per unit time is $\sum_i \alpha \nu_i \Gamma_i$. Our goal is to find the optimal lane configuration, i.e., the number of lanes to open for each payment type in order to minimize the total wait costs for all drivers.

$$\text{Minimize} \quad C = \sum_i \alpha \nu_i \left(\frac{\nu_i(\eta_i^2 \tau_i^2 + 1)}{2\eta_i(\chi_i \eta_i - \nu_i)} \right) \tag{2}$$

2) *Lower and Upper bounds Constraints*: Lanes to open for each collection type have lower and upper bounds, i.e., ξ_i and ζ_i. The lower bound ξ_i is determined for each i as in the previous section, and the number of ACM and ETC lanes available is limited by the upper bound, i.e., the number of lanes equipped with ACM and ETC lane machines.

$$\xi_i \leq \chi_i \leq \zeta_i \text{ for all } i \tag{3}$$

3) *Stability Constraints*: Next, enough toll booths are required so that the arrival rate to each lane should be less than the service rate for the system not to have unlimited queues. This condition guarantees that the queueing system will remain in the steady state with finite queues.

$$\nu_i/\eta_i < \chi_i \text{ for all } i \tag{4}$$

4) *Capacity Constraints*: Sum of open lanes for all lane types cannot be greater than the total available lanes. During the rush hours full capacity operation may be desired to minimize the total wait costs while partial operation should be fine during non-peak times as long as the service standard is met.

$$\sum_i \chi_i \leq \Omega \tag{5}$$

5) *Integer Constraints*: Obviously, the number of lanes to open should be an integer value. Non-negativity constraints are not needed because of the lower bounds constraints.

$$\chi_i = \text{int} \quad \text{for all } i. \tag{6}$$

Now let's consider the situation during the off-peak hours. During the off-peak times it may be needed to minimize the number of lanes to open for each lane type as long as the service level ψ_i is satisfied in order to save operation costs. In order to find the optimal lane configuration at the toll plaza during the off-peak hours given several constraints, we also devise a mathematical programming model to achieve this goal.

6) *Objective Function*: The operating cost is directly related with the number of lanes open at the toll plaza. Let β_i denote the operation cost of a type i lane per unit time. Then, the total operating costs B at the toll plaza per unit time become $\sum_i \beta_i \chi_i$. In this case, the objective function (2) should be replaced with the following.

$$\text{Minimize} \quad B = \sum_i \beta_i \chi_i \tag{7}$$

7) *Service Level Constraints*: We need to consider the requirement for the prescribed service level $\Gamma_i \leq \psi_i$ for all i, and, hence, add the following one in addition to the constraints (3)-(6).

$$\frac{\nu_i(\eta_i^2 \tau_i^2 + 1)}{2\eta_i(\chi_i \eta_i - \nu_i)} \leq \psi_i \text{ for all } i \tag{8}$$

Inequality (8) states that mean waiting time for each lane should not be greater than the prescribed service level ψ_i. For instance, the prescribed service standard for the ACM lane could be set to 30 seconds.

4 Model Applications

This section presents a case study finding time-dependent optimal lane configuration. We apply the proposed methodology with real traffic data which have been collected at the location of a potential toll plaza on the highway. The purpose of this section is to provide procedures of finding time-dependent optimal lane configuration to minimize waiting and operating costs.

After checking geographical feasibility, we find it is possible to design a toll plaza whose capacity Ω could be up to 16 lanes for one direction. In this case we consider

$T = 3$ dedicated lane types - ETC, ACM, and manual lanes - that may be representative in many situations. Toll plaza officials are interested in determining χ_i the appropriate number of lanes to open for each payment type $i = E, A, M$, during each one hour period in order to minimize the total wait cost of the motorists during the rush hours and in order to minimize operation cost during the off-peak times, given that the predetermined service level, ψ_i, is satisfied.

First, we obtain ν, the mean total arrival rate of vehicles for each one hour period, by measuring the traffic volume passing through a location of a potential toll plaza on the highway. However, exact values of ν_E, ν_A, and ν_M may not be readily available yet. As a result, we may obtain the proportions of motorists in other nearby toll plazas, interpolate and extrapolate those proportions to estimate values of θ_E, θ_A, and θ_M at this potential toll plaza and use the relationship $\nu_i = \nu\theta_i$ for $i = E, A, M$.

Figure 1 shows total mean arrival rate, ν, and mean arrival rates for each payment type, ν_E, ν_A, and ν_M. These rates may be location specific, but the increasing and decreasing trend of the traffic volume during a day in Figure 1 may be typical in many large cities in terms of the peak volume during the morning and evening peak hours and the low volume during the off-peak times. From Figure 1, it is generally known that morning rush hours are from 6:00 to 9:00 and evening rush hours are from 17:00 to 21:00. For instance, $\nu_E = 24.2$ between 8:00 and 9:00 states that on average 24.2 vehicles per minute approach the toll plaza between 8:00 and 9:00 to make toll payments through the ETC lanes.

Actual data on service times have been collected from a nearby toll plaza where the same ETC and ACM equipment is installed. We have mean service

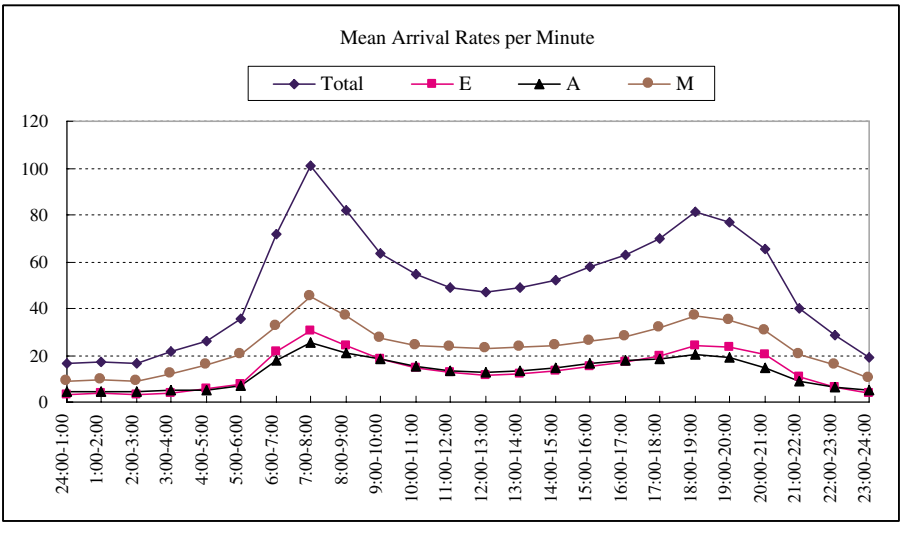


Fig. 1. Mean arrival rates, ν, ν_E, ν_A, ν_M, per minute

rates $\eta_E = 49.8,\ \eta_A = 10.2,\ \eta_M = 6.1$ per minute, and standard deviations of service times are $\tau_E = 0.01, \tau_A = 0.06, \tau_m = 0.12$ minutes for each lane.

Next, we determine the appropriate capacity of the new toll plaza. Lower bounds ξ_i are computed for each payment type over time as in the previous section. We have $\sum_i \xi_i = 12$ between 7:00 and 8:00, when we have the peak traffic volume during a day. Therefore, as long as the capacity of the toll plaza is set to greater than or equal to 12 toll lanes, mean arrival rate will be smaller than the mean service rate for each lane type and the new toll plaza may be able to remain in a steady state with finite queues. Although the lower bound for capacity of the toll plaza is only 12 during the peak time between 7:00 and 8:00, toll plaza officials set the capacity equal to 16, the maximum feasible number under the geographical restriction, considering the increasing trend of traffic volume in the future and correspondingly decide the upper bounds for each lane type.

Table 1. Lower and Upper Bounds

time	ξ_E	ζ_E	ξ_A	ζ_A	ξ_M	ζ_M	time	ξ_E	ζ_E	ξ_A	ζ_A	ξ_M	ζ_M
24-1	1	4	1	6	2	3	12-13	1	4	2	6	4	8
1-2	1	4	1	6	2	3	13-14	1	4	2	6	4	8
2-3	1	4	1	6	2	3	14-15	1	4	2	6	4	8
3-4	1	4	1	6	3	4	15-16	1	4	2	6	5	8
4-5	1	4	1	6	3	5	16-17	1	4	2	6	5	8
5-6	1	4	1	6	4	5	17-18	1	4	2	6	6	11
6-7	1	4	2	6	6	11	18-19	1	4	3	6	7	11
7-8	1	4	3	6	8	11	19-20	1	4	2	6	6	11
8-9	1	4	3	6	7	11	20-21	1	4	2	6	5	11
9-10	1	4	2	6	5	8	21-22	1	4	1	6	4	5
10-11	1	4	2	6	4	8	22-23	1	4	1	6	3	4
11-12	1	4	2	6	4	8	23-24	1	4	1	6	2	3

Although we have $\xi_E = 1$ all the time in Table 1, toll plaza officials would like to open at least two ETC lanes to encourage fast electronic toll payments to motorists. $\xi_A = 3$ between 7:00 and 8:00 implies that we need to install ACM equipment for at least three lanes. Once ACM equipment is installed, it may be better to use all three ACM lanes even during off-peak times as long as they do not have to be changed to other collection types since operating cost for ACM lanes is not significant. We now would like to know the optimal number of lanes to open for each lane type between these bounds to minimize waiting cost during the rush hours and to minimize operation costs during the off-peak times. Like ACM lanes, we mention that once the ETC machines are equipped, their operation costs are not considerable compared to the labor cost in the manual lanes.

We conduct a survey from the motorists in manual and ACM lanes and the results reveal that when the drivers at the toll plaza are in the queue up to 30 seconds, i.e., 0.5 minute, their perception of that waiting time in the queue is

only a few seconds. However, after waiting time in the queue becomes over 30 seconds, motorist perception of that waiting time grows quickly. Therefore, we set 30 second wait time as the prescribed service level. Toll plaza officials want ETC users not to stop to pay the toll in order to guarantee fast transaction with no waiting time although vehicles may have to decelerate as they approach the toll plaza, and we set $\psi_E = 2$ second $= 0.033$ minute.

Now we use mathematical programming models explained in the previous section to obtain the time-dependent optimal lane configuration. We utilize *MATLAB* and *Microsoft EXCEL* to conduct this numerical evaluation. Table 2 provides results by solving the mathematical programming models. For instance, by operating two ETC, three ACM and five manual lanes between 12:00 and 13:00 we can achieve minimum total waiting costs, which is directly proportional to the sum of wait times of all motorists approaching the toll plaza between 12:00 and 13:00, and mean wait times in the queue for each lane type are 0, 0.05, and 0.37 minutes per driver, respectively. It is noted that it is optimal to open only two ETC lanes, equal to the lower bound, throughout a day.

Table 2. Optimal Lane Configuration and Average Wait Times

time	χ_E^*	χ_A^*	χ_M^*	Γ_E	Γ_A	Γ_M	Γ	time	χ_E^*	χ_A^*	χ_M^*	Γ_E	Γ_A	Γ_M	Γ
24-1	2	3	2	0	0.01	0.37	205	12-13	2	3	5	0	0.05	0.37	536
1-2	2	3	2	0	0.01	0.40	228	13-14	2	3	5	0	0.05	0.42	637
2-3	2	3	2	0	0.01	0.34	185	14-15	2	3	5	0	0.06	0.48	755
3-4	2	3	3	0	0.01	0.26	195	15-16	2	3	6	0	0.08	0.32	584
4-5	2	3	4	0	0.01	0.23	218	16-17	2	3	6	0	0.09	0.41	790
5-6	2	3	5	0	0.02	0.25	321	17-18	2	4	10	0	0.06	0.14	325
6-7	2	3	11	0	0.09	0.12	328	18-19	2	4	10	0	0.07	0.19	514
7-8	2	4	10	0.01	0.11	0.36	1149	19-20	2	3	11	0	0.11	0.14	415
8-9	2	4	10	0	0.07	0.19	512	20-21	2	3	11	0	0.06	0.10	250
9-10	2	3	6	0	0.10	0.35	685	21-22	2	3	5	0	0.03	0.24	310
10-11	2	3	5	0	0.07	0.49	783	22-23	2	3	4	0	0.02	0.22	216
11-12	2	3	5	0	0.05	0.40	601	23-24	2	3	3	0	0.01	0.17	107

We realize that during the off-peak hours it may not be required to operate all the lanes, but prescribed service standard is quite satisfied. For example, although only two ETC, three ACM, and two manual lanes are utilized between 2:00 and 3:00, which means only 7 lanes are used out of 16, mean waiting time in each lane is still below the specified service levels. During this time period one ETC lane and one ACM lane may be fine to satisfy the predetermined service level, but it may not be attractive to further reduce the number of open lanes for ETC and ACM lanes since their operation costs are trivial compared to the labor costs in the manual lanes and they do not have to be changed to the manual lanes, either.

Table 2 reveals that the current 16 lane toll plaza system may have enough capacity to hold the current traffic volume with some slack capacity. During the rush hours in the morning and in the evening we can keep the mean waiting times for each lane type below the prescribed service level with full capacity

operation. In addition, service level is still well maintained during the off-peak hours by using partial capacity.

5 Conclusion

We have presented mathematical models for the time-dependent optimal lane configuration policy at the toll plaza. With the $M/G/1$ queueing process, which is both realistic and mathematically manageable for the toll plaza operations, we computed lower bounds on the number of required lanes to open for each payment type, and we could determine the total capacity of the toll plaza. Then, in order to find time-dependent optimal lane configuration we comined the proposed queueing process into mathematical programming in each time period. We then illustrated how the introduced methodology could be applied in practice for time-dependent optimal lane configuration policy with a case study based on the actual traffic data collected. By numerical evaluation, we showed that ETC system is powerful to reduce long waits and increase the throughput at the toll plaza.

Although the author have used only three representative dedicated lane types (ETC, ACM, and manual lanes) in the case study, the proposed methods and results are extendable to the case when $T > 3$ with any other payment types or any mix of dedicated lane types. By considering development in electronic display technology for time-dependent lane configuration and evolution of computer ability, we remark that implementation of the proposed methodology is not difficult with small amount of computational efforts. To completely reflect seasonal factors or other changes in traffic volume, we need to regularly update the database of total traffic volume, proportion of lane users, etc.

The case study in Section 4 showed how the introduced mathematical models could be applied in operating a toll plaza based on the *current* traffic volume. However, the total traffic volume in many metropolitan areas may continuously increase up to some level in the future. Hence, when designing a toll plaza, we also need to forecast the increase of future traffic volume, e.g., the percent increase per year, and examine corresponding effects on the mean wait times by the sensitivity analysis. If the increase in future traffic volume is considerable, the currently designed toll plaza may not be efficient or unable to handle the high traffic volume during rush hours. In case expanding the toll plaza is not possible due to geographical restrictions, we may need to devise a strategy, for example, by transferring toll users from low-throughput to high throughput lanes in order to resolve congestion. Calculation of the minimum required transfer amount and numerical evaluation of that kind of strategy to see how effective it is in reducing total wait times will be interesting and practical future research topics. In addition, we need to consider how to implement the strategy, for example, with some discount benefits or advertising.

Although we obtained the optimal lane configuration based on the historical average traffic volume data, real-time traffic volume or real-time proportion of toll users may be considerably different from the historical average values,

contingent on traffic accidents, weather conditions, sports events, etc. Thus, in order to best use the introduced methodologies in the time-dependent optimal lane configuration, we need to combine the historical and real-time traffic information. When we have these unexpected changes of the total traffic volume or proportion of toll users, how to dynamically reconfigure the toll plaza in real-time to maintain not long wait time is another future research direction as introduced with decision support system in [22].

Lastly, finding the optimal work force scheduling with full-time and part-time workers to minimize operating cost based on stochastic traffic situations may be an interesting but challenging future research topic. It may be required to hold some additional toll collectors as a buffer against unexpected surge of the toll demand. In that situation, forecasting the future traffic volume should be crucial to minimize labor costs of buffer toll collectors. The optimal scheduling of full-time and part-time toll collectors may be complex in practical situations if it has to consider all aspects of meal time, break and rest, day off, etc.

References

1. Heyman, D.: Optimal operating policies for $M/G/1$ queueing systems. Journal of the Operational Research Society of America 16, 362–382 (1968)
2. Blackburn, J.: Optimal control of a single server queue with balking and reneging. Management Science 19, 297–313 (1972)
3. Mendelson, H.: Pricing computer services: queueing effects. Communications of the ACM 28, 312–321 (1985)
4. Bell, C.: Optimal operation of an $M/M/2$ queue with removable servers. Operations Research 28, 1189–1204 (1980)
5. Szarkowicz, D.: Optimal control of an $M/M/S$ queueing system. Operations Research 33, 644–660 (1985)
6. Boronico, J., Siegel, P.H.: Capacity planning for toll roadways incorporating consumer wait time costs. Transportation Research Part A 32, 297–310 (1998)
7. Leeman, W.: The reduction of queues through the use of prices. Operations Research 12, 783–785 (1964)
8. Kay, J.: Uncertainty, congestion and public utility pricing. Review of Economic Studies 46, 601–611 (1979)
9. Chao, H.P.: Peak-load pricing and capacity planning with demand and supply uncertainty. Bell Journal of Economics 14, 179–190 (1983)
10. Arnott, R., Palma, A.D., Lindsey, R.: A structural model of peak-period congestion: a trac bottleneck with elastic demand. The American Economic Review 83, 161–179 (1993)
11. Sullivan, R.: Fast lane. Forbes 154, 112–115 (1994)
12. Burris, M., Hildebrand, E.: Using microsimulation to quantify the impact of electronic toll collection. ITE Journal 66, 21–24 (1996)
13. Al-Deek, H.M., Mohamed, A.A., Radwan, A.E.: Operational benefits of electronic toll collection: case study. Journal of Transportation Engineering 123 (1997)
14. Zarrillo, M.L., Radwan, A.E., Al-Deek, H.M.: Modeling Traffic Operations at Electronic Toll Collection and Traffic Management Systems. Computers and Industrial Engineering 33, 857–860 (1997)

15. Li, J., Gillen, D., Dahlgren, J., Mekky, A.: Benefit-cost evaluation of the electronic toll collection system: A comprehensive framework and application. Transportation Research Record 1659, 31–38 (1999)
16. Levinson, D., Chang, E.: A model for optimizing electronic toll collection systems. Transportation Research Part A 37, 293–314 (2003)
17. Klodzinski, J., Al-Deek, H.M.: Proposed Level of Service Methodology for a Toll Plaza. Transportation Research Record 1802, 86–96 (2002)
18. Hasofer, A.: On the single-server queue with non-homogeneous Poisson input and general service time. Journal of Applied Probability 1, 369–384 (1964)
19. Schwartz, B.: Queueing models with lane selection: A new class of problems. Operations Research 22, 331–339 (1974)
20. Hillier, F., Lieberman, G.J.: Introduction to operations research. McGraw-Hill, New York (2005)
21. Liu, W.-M., Li, R.: Optimization method of toll discount rate in electronic toll collection systems. Journal of South China University of Technology (Natural Science) 33, 63–68 (2005)
22. Zarrillo, M.L.: Using decision support system and graphical user interface to choose appropriate lane configurations at toll facilities. Transportation Research Record 1935, 8–16 (2005)

Author Index