Andrzej Kwiecień
Piotr Gaj
Piotr Stera (Eds.)

Communications in Computer and Information Science 39

# Computer Networks

16th Conference, CN 2009
Wisła, Poland, June 2009
Proceedings

Springer

Communications
in Computer and Information Science 39

Andrzej Kwiecień Piotr Gaj
Piotr Stera (Eds.)

# Computer Networks

16th Conference, CN 2009
Wisła, Poland, June 16-20, 2009
Proceedings

Springer

Volume Editors

Andrzej Kwiecień
Piotr Gaj
Piotr Stera

Silesian University of Technology
Institute of Informatics
Gliwice, Poland

E-mail: {andrzej.kwiecien, piotr.gaj, piotr.stera}@polsl.pl

Library of Congress Control Number: Applied for

CR Subject Classification (1998): C.2, D.4.4-5, H.3.4-5, K.4.4

| | |
|---|---|
| ISSN | 1865-0929 |
| ISBN-10 | 3-642-02670-2 Springer Berlin Heidelberg New York |
| ISBN-13 | 978-3-642-02670-6 Springer Berlin Heidelberg New York |



springer.com



Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12707454 06/3180 5 4 3 2 1 0

# Preface

The continuous and very intense development of IT has resulted in the fast development of computer networks. Computer networks, as well as the entire field of IT, are subject to constant changes triggered by the general technological advancement and the influence of new IT technologies. These methods and tools of designing and modeling computer networks are becoming more advanced. Above all, the scope of their application is growing thanks to, for example, the results of new research and because of new proposals of application, which not long ago were not even taken into consideration. These new applications stimulate the development of scientific research, as the broader application of system solutions based on computer networks results in a wide range of both theoretical and practical problems. This book proves that and the contents of its chapters concern a variety of topics and issues.

Generally speaking, the contents can be divided into several subject groups. The first group of contributions concerns new technologies applied in computer networks, particularly those related to nano, molecular and quantum technology. Another group comprises chapters devoted to new technologies related to computer network structure. There are also chapters concerning the fundamentals of computer networks, their architecture and programming. A very important group of chapters concerns the Internet in its broad meaning, covering a variety of topics. The next group comprises papers related to data and analysis of industrial computer networks. This issue is also of fundamental importance from the point of view of the structure and industrial application of distributed real-time IT systems. The last group of contributions describes general applications of computer networks, including issues related to the quality of data exchange. On behalf of the Program Committee, we would like to take this opportunity to express our gratitude to all the authors for sharing the results of their research and for assisting in the creation of this book, which in our view is a valuable bibliographic source within the area of computer networks.

April 2009

Andrzej Kwiecień
Piotr Gaj

# Organization

CN 2009 was organized by the Institute of Informatics, Silesian Univeristy of Technology (SUT) at support by the Committee of Informatics of the Polish Academy of Sciences, Section of Computer Network and Distributed Systems in technical cooperation with the Polish Section of IEEE.

Institute of Informatics
Silesian University of Technology
ul. Akademicka 16
44-100 Gliwice, Poland
e-mail: siecikomputerowe@polsl.pl
web: http://cn.polsl.pl

## Executive Committee

All members of the Executing Committee are from the Silesian University of Technology, Poland.

| | |
|---|---|
| Honorable Member | Halina Węgrzyn |
| Organizing Chair | Piotr Gaj |
| Technical Volume Editor | Piotr Stera |
| Technical Support | Aleksander Cisek |
| Technical Support | Arkadiusz Jestratjew |
| Office | Małgorzata Gładysz |
| Web Support | Piotr Kuźniacki |
| IEEE Coordinator | Jacek Izydorczyk |

## Program Committee

### Program Chair

| | |
|---|---|
| Andrzej Kwiecień | Silesian University of Technology, Poland |

### Honorary Members

| | |
|---|---|
| Klaus Bender | TU München, Germany |
| Zdzisław Duda | Silesian University of Technology, Poland |
| Andrzej Karbownik | Silesian University of Technology, Poland |
| Jerzy Rutkowski | Silesian University of Technology, Poland |
| Stefan Węgrzyn | IITiS PAN Gliwice, Poland |
| Bogdan M. Wilamowski | Auburn University, USA |

### Program Committee Members

| | |
|---|---|
| Tülin Atmaca | Institut National de Télécommunication, France |
| Win Aung | National Science Foundation, USA |
| Leszek Borzemski | Wrocław University of Technology, Poland |
| Markus Bregulla | University of Applied Sciences Ingolstadt, Germany |
| Tadeusz Czachórski | Silesian University of Technology, Poland |
| Andrzej Duda | INP Grenoble, France |
| Jean-Michel Fourneau | Université de Versailles, France |
| Natalia Gaviria | Universidad de Antioquia, Colombia |
| Jerzy Klamka | IITiS PAN Gliwice, Poland |
| Demetres D. Kouvatsos | University of Bradford, UK |
| Stanisław Kozielski | Silesian University of Technology, Poland |
| Henryk Krawczyk | Gdańsk University of Technology, Poland |
| Nihal Pekergin | Université de Versailles, France |
| Piotr Pikiewicz | Academy of Business, Poland |
| Bolesław Pochopień | Silesian University of Technology, Poland |
| Frank Schiller | TU München, Germany |
| Mirosław Skrzewski | Silesian University of Technology, Poland |
| Sylwester Warecki | Freescale Semiconductor, USA |
| Tadeusz Wieczorek | Silesian University of Technology, Poland |
| Bane Vasic | University of Arizona, USA |
| Grzegorz Zaręba | University of Arizona, USA |

## Referees

Tülin Atmaca
Leszek Borzemski
Markus Bregulla
Tadeusz Czachórski
Natalia Gaviria
Henryk Krawczyk
Jerzy Klamka
Stanisław Kozielski
Andrzej Kwiecień
Bolesław Pochopień
Piotr Pikiewicz
Frank Schiller
Mirosław Skrzewski
Sylwester Warecki
Tadeusz Wieczorek
Bene Vasic
Grzegorz Zaręba

## Sponsoring Institutions

Technical cosponsor of the conference: Polish Section of IEEE.

# Table of Contents

# Molecular Networks and Information Systems

Stefan Węgrzyn[1] and Lech Znamirowski[2]

[1] Polish Academy of Sciences, Gliwice
[2] Silesian University of Technology, Institute of Informatics
stefan.wegrzyn@polsl.pl

**Abstract.** Technical systems of informatics are discussed, such that their structure is based on molecules, similarly to the biological systems of informatics. A path to such technical systems of informatics is pioneered by the currently led works in the field of molecular genetic engineering.

## 1 Introduction

Molecular information systems are defined as the information systems whose structures are based directly on molecules, their properties and bonds between them. The molecules are groups of atoms held together by bonds. Spatial arrangement of atoms composing a molecule is defined as the molecular structure. It is a three-dimensional structure and therefore its representation on the two-dimensional plane of text requires the use of any of the representation methods used in such cases. The methods most often used are: space-filling, ball, skeletal, ribbon and surface representations. According to the international agreement the following colours have been selected to represent individual atoms on those representations: carbon – black, oxygen – red, hydrogen – white, sulphur – yellow, nitrogen – blue, and phosphorus – purple.

For example in (Fig. 1) different space-filling representations of molecular structures of: water, acetate and formamide are presented according to [1].

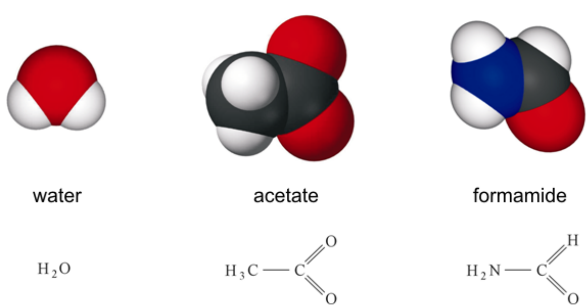



**Fig. 1.** Structural patterns and space-filling representations of the molecules of: water, acetate and formamide

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 1–10, 2009.

Although stereochemical molecular structures illustrate a molecular structure very well, they are difficult to draw and require colouring.

An alternative method for representing structures involves the use of Fisher projections in the form of horizontal and vertical lines that connect e.g. substituent atoms with the carbon atom, which is located at the centre of the cross. It is assumed that the horizontal bond is above the page and the vertical bond below the page (Fig. 2).

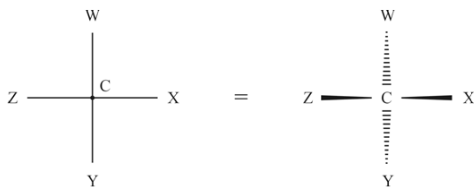



**Fig. 2.** The Fisher projections. The Z and X atoms are located above the page plane and the W and Y atoms below that plane.

The following definitions are used in the analysis of molecular geometry: conformation, configuration and chiral objects. They are described shortly below.

A conformation is defined as a way of arrangement of atoms in a molecule around a single atom as in the example of the methane ($CH_4$) molecule presented in Fig. 3.

If there are several groups of atoms bonded together in a molecule, the bonds between those groups are defined as a configuration. A chiral molecule is the molecule which cannot be superposed on its mirror image. The name chiral molecule comes from the Greek word *cheir* = hand, as it is illustrated in Fig. 4.

A helix is also a chiral object (Fig. 5).

The molecular structure analysis, and especially changes which may occur in the structure and the influence which this may have on the parameters such as freezing temperature, is of particular importance in view of the Mpemba effect illustrated in Fig. 6.

Analyses of spatial structures of molecules and dynamic processes occurring in them gave rise to the development of the science divisions such as nanochemistry, femtochemistry and even attochemistry.

$$1\,\mathrm{nm} = 10^{-9}\,\mathrm{m} \qquad 1\,\mathrm{fsec} = 10^{-15}\,\mathrm{sec} \qquad 1\,\mathrm{asec} = 10^{-18}\,\mathrm{sec} \tag{1}$$

The molecular technologies of object construction are to be understood as such object synthesis methods which involve inserting free molecules transferred from available external environment into the structures being constructed. An example of such molecular information technologies can be the technologies

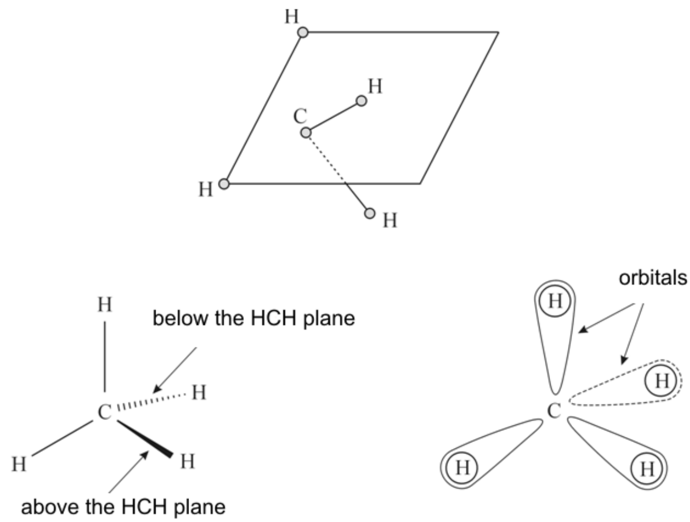



**Fig. 3.** Conformations of hydrogen atoms in the methane ($CH_4$) molecule

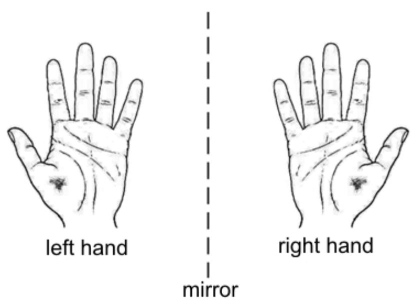



**Fig. 4.** The name chiral object comes from the Greek word *cheir* – hand

being the basis for formation and development processes of information systems which have been continued in biological organisms for billion of years, owing to which those organisms were and are formed, were and are developed and are reproduced.

The molecular information technologies can only be developed in the appropriately prepared environments. The following can be specified as the necessary environmental conditions:

– motion of environmental molecules ensured by molecular motors (Fig. 7), based on controlled switches turning alternately positive feedbacks into negative ones and vice versa.

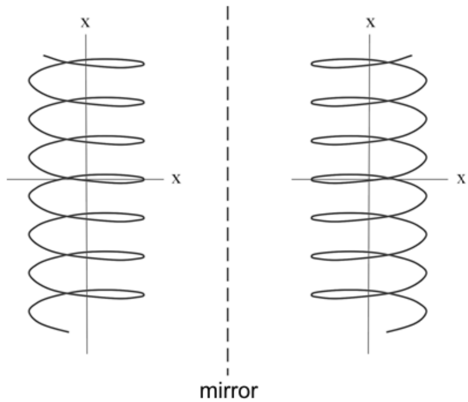



**Fig. 5.** A helix as an example of chiral object

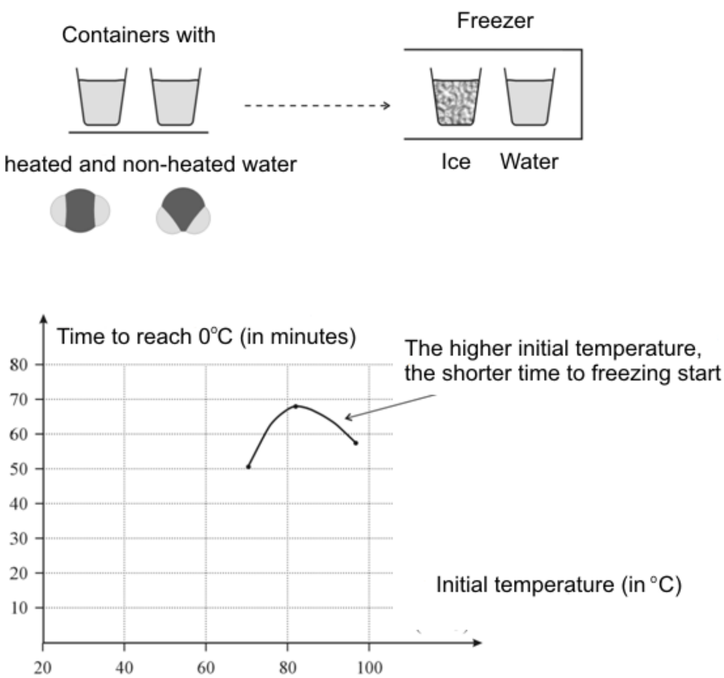



**Fig. 6.** The Mpemba effect

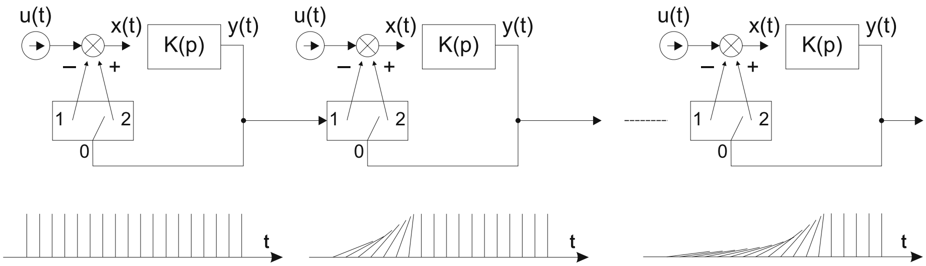



**Fig. 7.** Structure of molecular motors. State motion.

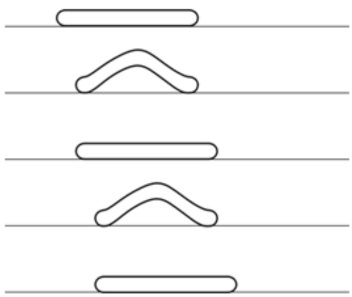


**Fig. 8.** How motion was created in the evolution processes. First forms of motion – crawling of single cell organisms.

Motion of elements is the basic condition of molecular technologies. The motion formation originates from the period when the first living organisms were single cells. A single cell organism was gaining motion as a result of a change of its shapes as illustrated in Fig. 8.

One more condition that should be met by environment is:

– enclosing the environment by means of membranes to protect against unwanted diffusions of molecules into some of its areas (Fig. 9 and Fig. 10).

Figure 9 illustrates a two-layer membrane structure and symbol of a molecule called lipid which the membrane structure is based on.

A simplified two-layer lipid membrane can be presented in the way illustrated in (Fig. 10).

In the environment prepared like that, the molecular technology of connecting molecules and atoms into the needed products can be developed. Its essence is such a notation form of the encoded information that, when inserted into the disordered group of molecules, results in creation of ordered subgroups, being the expected products, in it (Fig. 11).

One of the ways leading to that is the sticky matrix method presented in Fig. 12.

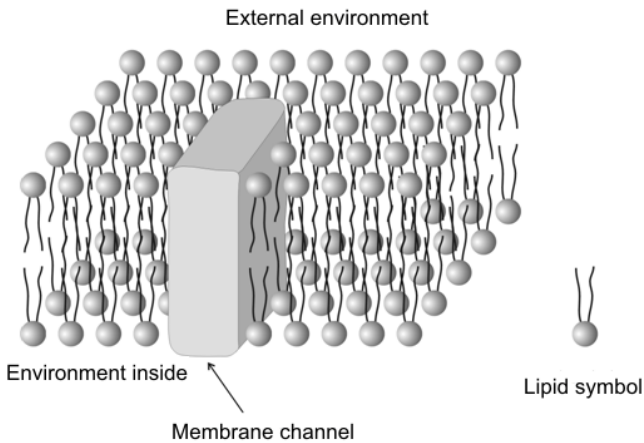



**Fig. 9.** Fragment of two-layer membrane based on the molecules called lipids and symbol of that molecule

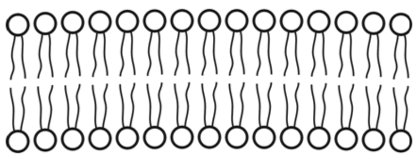


**Fig. 10.** Schematic image of lipid bilayer. Such structures define the boundaries of some areas inside an environment.

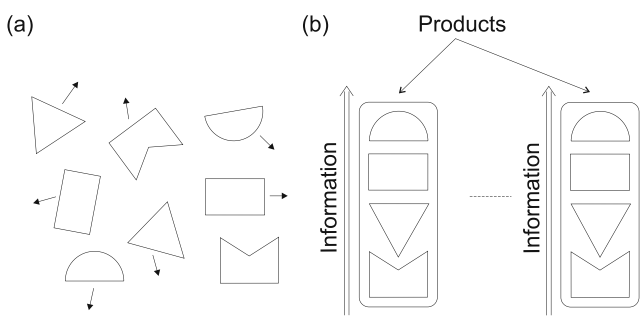



**Fig. 11.** (a) Disordered group of molecules, (b) ordered subgroups creating the products owing to information input

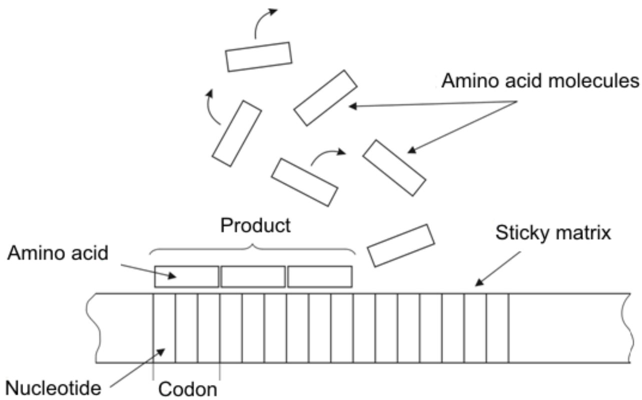



**Fig. 12.** Illustration of sticky matrix method in the case of protein synthesis

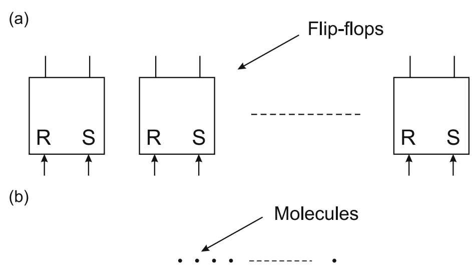



**Fig. 13.** Illustration of text forms in the: (a) technical and (b) molecular information systems

Illustration of text structures in the molecular information systems in comparison with the technical information systems is presented in Fig. 13.

The molecular algorithm notations presented in Fig. 13 are chains of different molecules occupying thousands of times less space than algorithm notations in the technical information systems. The texts noted like that are located in the DNA of living organisms being the basis of their existence, development and reproduction.

An encoded notation of product structures is presented in Fig. 14, and the whole molecular information system in Fig. 15.

An example of molecular information technologies can be the biological information systems being the basis of existence, development and reproduction of living organisms. They are the basis for some solutions of the processes performed within genetic engineering for example insulin production. They are described shortly below.

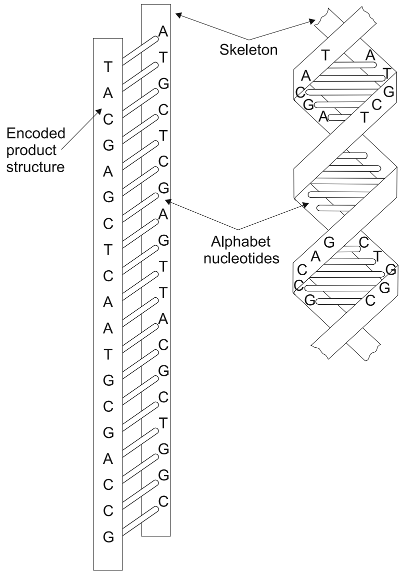



**Fig. 14.** Packed notation of a product structure in the molecular information system

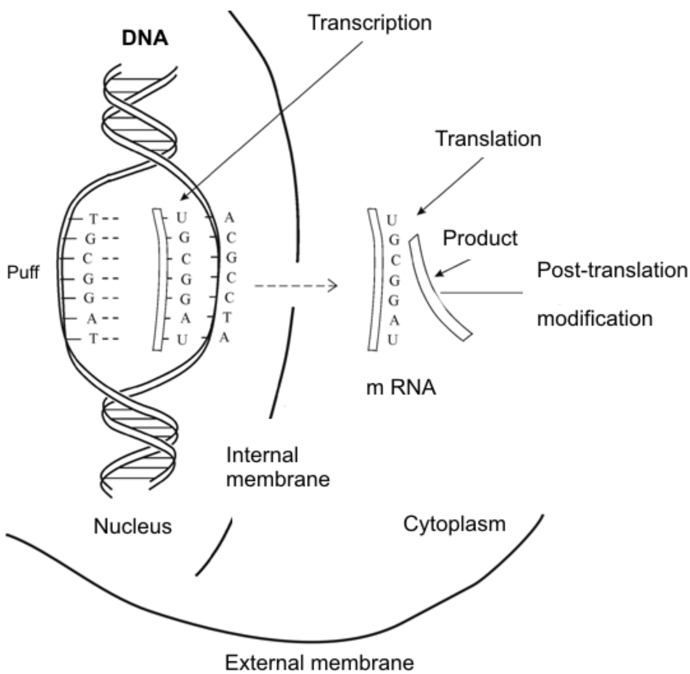



**Fig. 15.** Structure of the molecular information system

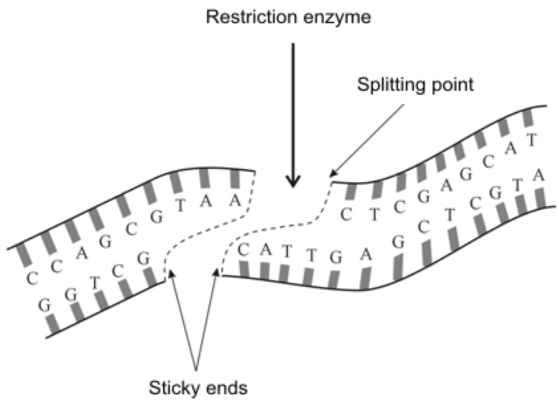



**Fig. 16.** Illustration of cutting the DNA chain

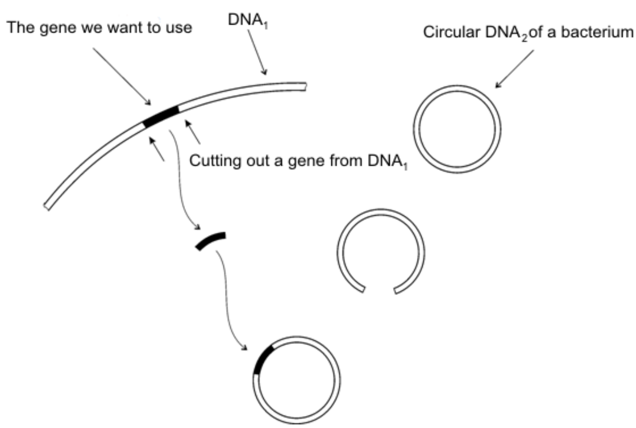



**Fig. 17.** Cloning within DNA2 of a gene cut out of DNA1

## 2 Molecular Genetic Engineering

The first operation in genetic engineering is cutting the DNA chain as illustrated in Fig. 16.

The DNA chain, chemically cut by means of restriction enzymes, can be the basis for next operations which we want to perform e.g. cloning as presented in Fig. 17 and Fig. 18.

## 3 Compendium

A molecular environment is understood as a group of different, free and able to move atoms and molecules, which are defined as its elements.

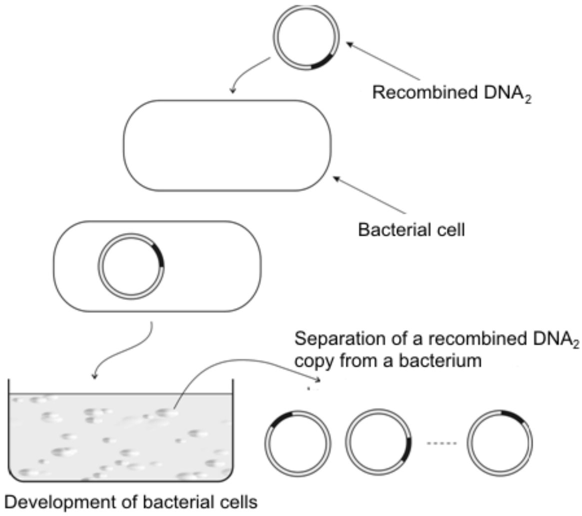



**Fig. 18.** Cloning of alien structural genes in guest cells

In such a molecular environment, different local bonds between their elements can be created.

The local bonds, whose structures store the information on history of their formation, can be molecular origins of living objects that is such objects, which by their existence, generate formation processes of next derivative objects imitated on them.

We can then say about origins of living objects and methods of molecular storage and transfer of information, which they are based on, are molecular information systems.

## References

1. Berg, J.M., Tymoczko, J.L., Stryer, L.: Biochemistry. PWN, Warszawa (2007)
2. Węgrzyn, S.: Nanosystem of informatics. International Journal of Systems Science 32(12) (2001)
3. Ball, P.: Does hot water freeze first? Physics World 19(4) (April 2006)

# Sorting of Quantum States with Respect to Amount of Entanglement Included

Roman Gielerak and Marek Sawerwain

Institute of Control & Computation Engineering
University of Zielona Góra, ul. Podgórna 50, Zielona Góra 65-246, Poland
{R.Gielerak,M.Sawerwain}@issi.uz.zgora.pl

**Abstract.** The canonical Schmidt decomposition of quantum states is discussed and its implementation to the Quantum Computation Simulator is outlined. In particular, the semiorder relation in the space of quantum states induced by the lexicographic semiorder of the space of the Schmidt coefficients is discussed. The appropriate sorting algorithms on the corresponding POSETs (partial ordered sets) consisting from quantum states are formulated and theirs computer implementations are being tested.

## 1 Introduction

Let $\mathcal{E}(H_A \otimes H_B)$ be a set of quantum states of a given composite system $S_{A+B}$ on the corresponding Hilbert space $H_A \otimes H_B$. As it is well known the problem to decide whether a given state $\rho \in \mathcal{E}$ is entangled or not is in general NP-HARD [5]. Although some mathematical procedures for this purpose and also for deriving in a quantitative way the amount of entanglement included do exist the real problem with them is that they are hardly to be efficiently calculable [2] and [3].

In the case of pure states the operational semiorder $|\psi\rangle \overset{LOCC}{\succ} |\varphi\rangle$ on the space of pure states (meaning that the state $|\psi\rangle$ can be transformed into the vector $|\varphi\rangle$ by exclusive use of only LOCC (local operations and classical communication) class of operations) has been formulated in an effectively calculable way by appealing to the corresponding Schmidt decomposition in Nielsen [4]. The relation $\overset{LOCC}{\succ}$ introduces only partial order on the space of pure states and it is why the adaptations of standard sorting algorithms of the corresponding POSETs are much more involved [6] and [7] comparing to the case of linearly ordered sets. Several adaptations of this sorting procedures are being adopted and tested on the Zielona Góra Quantum Computing Simulator and some recent results of this kind will be presented in the present contribution, see also [7].

Another topic discussed in our contribution is an attempt to generalize the Nielsen result to the case of general mixed quantum states. For this purpose the Schmidt decomposition in the corresponding Hilbert-Schmidt space has been used and certain semiorder relation in the space of quantum states has been

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 11–18, 2009.

introduced. The effort has been made to connect the introduced semiorder with several notions of quantitative measures of entanglement.

A well known distillation of entanglement procedure [1] also introduces a partial semiorder on the space of quantum states. However this process is hardly to be effectively calculable and moreover it requires to have many (infinitely many in fact) copies of a given unknown quantum state at hands in order to perform the distillation process.

Although we have no complete proof we formulate a conjecture that the partial order induced by the lexicographic order of the Schmidt decomposition coefficients is connected to the operational meaning saying that in the state $\rho_1$ is no less entanglement contained then in the state $\rho_2$ if the state $\rho_1$ may be transformed into $\rho_2$ by means of local operations supplemented by classical communication only.

## 2 Algorithms for Sorting Quantum States

### 2.1 Canonical Schmidt Decomposition of Quantum States

For a given finite-dimensional Hilbert space $\mathcal{H}$ the corresponding Hilbert-Schmidt space is denoted as $HS(\mathcal{H})$. Let us recall that the space $HS(\mathcal{H})$ consist of all linear operations acting on $\mathcal{H}$ and equipped with the following scalar product:

$$\langle A|B\rangle_{HS} = \mathrm{Tr}\left(A^{\dagger}B\right) \tag{1}$$

A system $(E_i, i = 1, \ldots, \dim(\mathcal{H})^2)$ of linearly independent matrices on $\mathcal{H}$ is called complete orthonormal system iff $\langle E_i|E_j\rangle_{HS} = 1$. If moreover all $E_i$ are hermitean the system $(E_i)$ is called complete hermitean orthonormal system.

**Proposition 1.** *Let $\rho \in \mathcal{E}(\mathcal{H}_A \otimes \mathcal{H}_B)$. Then there exist: a number $r > 0$ (called the canonical Schmidt rank of $\rho$) and a complete orthonormal system $(E_i^A)$ (resp. $(E_j^B)$) in $HS(\mathcal{H}_A)$ (resp. $HS(\mathcal{H}_B)$) and such that*

$$\rho = \sum_{\alpha=1}^{r} \lambda_\alpha E_\alpha^A \otimes E_\alpha^B \tag{2}$$

*where the numbers $\lambda_\alpha > 0$ are called (the canonical) Schmidt coefficients of $\rho$. If all $\lambda_\alpha$ are different then this decomposition is unique.*

*Remark 1.* A different notions of Schmidt decomposition of density matrices are being discussed in the literature [9]. Our Schmidt characteristics like the canonical Schmidt rank and (canonical) Schmidt coefficients and the corresponding orthonormal systems are in unique way connected with a given $\rho$ and in principle all the properties (separability|nonseparability for example) of $\rho$ should be obtainable form this decomposition. For example if the corresponding $E_\alpha^A$, $E_\alpha^B$ in formula (2) are nonnegative and therefore hermitean then $\rho$ is separable.

*Remark 2.* It is well known [10], [11] that for separable states the sum of the canonical Schmidts coefficients is always less or equal to 1. This leads to the separability criterion known as cross norm criterion.

*Remark 3.* The closed subspace of $HS(\mathcal{H})$ consisting of hermitean matrices forms a real Hilbert space. Therefore if the SVD theorem extends to the real Hilbert space case then the corresponding systems in formula (2) are hermitean by the very construction.

Now we formulate constructive route to the canonical Schmidt decomposition.

**Proposition 2.** *Let $d = \dim(\mathcal{H})$ and let $(E_i, i = 1, \ldots, d^2)$ be a system of linear independent matrices on $\mathcal{H}$. Then there exists operation $\mathcal{O}$ converting the system $(E_i)$ into the orthonormal system $(F_i)$. If the system $(E_i)$ consists of hermitean matrices then $\mathcal{O}((E_i))$ is also system formed from hermitean matrices.*

*Proof.* The well known Gram-Schmidt orthonormalisation procedure is used as an example of the converting operation $\mathcal{O}$. ∎

Let now $(F_i^A)$ (resp. $(F_j^B)$) be any orthonormal system in $HS(\mathcal{H}_A)$ (resp. $HS$ $(\mathcal{H}_B)$). Then the system $(F_i^A \otimes F_j^B)$ forms a complete orthonormal system in $HS(\mathcal{H}_A \otimes \mathcal{H}_B)$. Thus taking any $\rho \in \mathcal{E}(\mathcal{H}_A \otimes \mathcal{H}_B)$ we can decompose:

$$\rho = \sum_{i,j=1} c_{i,j} F_i^A \otimes F_j^B \tag{3}$$

where $c_{i,j} = \mathrm{Tr}\left(\rho F_i^A \otimes F_j^B\right)$.

Then we apply SVD operation to the matrix $C = (c_{i,j})$ yielding (like in the vector case) all the data for supplying the decomposition (2). In particular the singular values of the matrix $C$ are equal to the squares of the Schmidt numbers from (2).

### 2.2 Linear and Partial Semi-order for Entanglement States

Firstly, we present a simple algorithm to realise sorting a set of quantum states by using von Neumann entropy notion. We will call this algorithm a linear sorting by entropy algorithm (abbreviated as LSEA). The pseudo-code of LSEA is presented in Fig. 1.

The second presented algorithm realises sorting of entangled states using the Schmidt decomposition. The pseudo-code is presented in the Fig. 4. The input of the Algorithm (Fig. 4) is now a list $\mathbb{V}$ of vector states on the space $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$.

**function** LSEA( $\Sigma : \{\rho_1, \rho_2, \ldots, \rho_N\}$ ) : $\Sigma^{SORT} : \{\rho_1, \rho_2, \ldots, \rho_N\}$
  **for** i=1 **to** N **do**
    $\rho_i^A = \mathrm{Tr}_{H_A}(\rho_i)$
    $[\sigma_i, V_i] = \mathrm{EigenSystem}(\rho_i^A)$
    $\mathrm{En}(i) = E(\rho_i^A) = -\sum_k \lambda_k^i \log \lambda_k^i$
  **end for**
  $\Sigma^{SORT} = \textsc{ClassicalSort}(\{\mathrm{En}(1), \mathrm{En}(2), \ldots, \mathrm{En}(N)\})$
  **return** $\Sigma^{SORT}$
**end function**

**Fig. 1.** Algorithm for sorting entangled quantum states using the von Neumann entropy

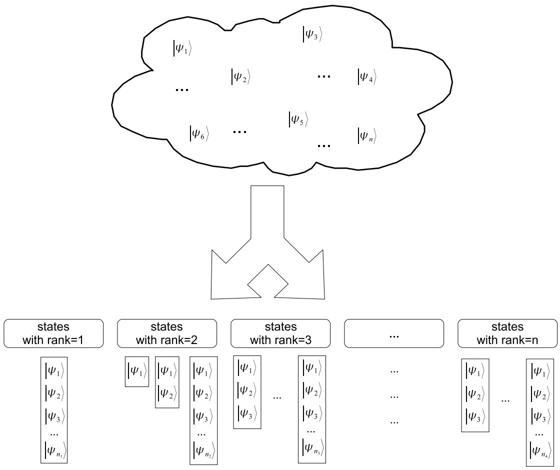



**Fig. 2.** General idea of sorting quantum states where the lexicographic order is used. In result the obtained structure represents the partial order where in the sets of quantum states with the same Schmidt rank may appear a linear chains which are non-comparable.

The output is divided into two parts. The first part is the partitioning of $\mathbb{V}$: $V = [V_1, \ldots, V_p]$ where

$$\begin{aligned} V(i) \in \mathbb{V}, \mathrm{SchmidtRank}(V_i) = r_i = const; \text{ and} \\ 1 \leq r_1 < r_2 < \ldots < r_p \leq \min(\dim \mathcal{H}_A, \dim \mathcal{H}_B),\ U_i V(i) = \mathbb{V} \end{aligned} \tag{4}$$

i.e. the partitioning with respect to increasing Schmidt's ranks. Additionally, we return the complete data of merging of each $V_i$.

We need to define the corresponding semiorder relation. For a given pair of quantum states $\rho_1$ and $\rho_2$ we perform the canonical Schmidt decomposition (2) of them obtaining the corresponding Schmidt ranks $r_1$, $r_2$ and the corresponding Schmidt coefficients $\lambda_\alpha(1)$, $\lambda_\alpha(2)$. Having this data we can formulate the following algorithm which in fact introduces the semiorder relation in the space of quantum states. This algorithm will be called SD-QueryOracle.

Having defined the semiorder $\prec$ we can now formulate one of the possible (see [12] for other versions) sorting sets of quantum states algorithm that will be called MergeSort type algorithm.

### 2.3 Example of Usage of Linear Sorting Algorithm

Let us consider family of Bell maximally entangled states for qubits and qudits. In the case of qubits, these states have the following form

$$|\psi\rangle = \alpha|00\rangle \pm \beta|11\rangle \quad \text{or} \quad |\psi\rangle = \alpha|01\rangle \pm \beta|10\rangle \quad \text{and} \quad |\alpha|^2 + |\beta|^2 = 1 \ , \tag{5}$$

**function** SD-QueryOracle( $V : \{\rho_1, \rho_2\}$ ) : $\{(\rho_1 \prec \rho_2), (\rho_2 \prec \rho_1), \text{non} - \text{comparable}\}$
  **for** i=1 **to** 2 **do** $\{r_i, \{\lambda_1(i), \ldots, \lambda_{r_i}(i)\}\} = \text{SchmidtDecomp}(V_i)$
  **if** $(r_1 > r_2)$ return $(\rho_1 \prec \rho_2)$
  **if** $(r_2 > r_1)$ return $(\rho_2 \prec \rho_1)$
  sort the sets $\{\lambda_1(i)_{i=1,2}\}$ in non-increasing order
  **if** $(r_1 = r_2)$ **if** $(\forall_{j=1,\ldots,r_1} \left(\sum_{i=1}^{j} \lambda_i^2(1) \leq \lambda_i^2(2)\right))$ return $(\rho_1 \prec \rho_2)$
  **return** non-comparable
**end function**

**Fig. 3.** Implementation of query oracle for sorting entangled quantum states

**function** ChainMergeSort( $\Sigma : \{\rho_1, \ldots, \rho_N\}$ ) : $V$
  **for** i=1 **to** N **do** $\{r(i), \{\lambda_1(i), \ldots, \lambda_{r_i}(i)\}\} = S(i) = \text{SchmidtDecomp}(V_i)$
  $r^\star = \max(r(1), \ldots, r(n))$
  **for** $\alpha$=1 **to** $r^\star$ **do**
    $V(\alpha) = [\,]$
    **for** i=1 **to** N **do if** (r(i)=$\alpha$) V($\alpha$)=[V($\alpha$), $\rho_\alpha$]
  **end for**
  **for** $\alpha$=1 **to** $r^\star$ **do**
    choice randomly $\rho \in V(\alpha)$
    $R^\alpha = (\rho, \{\,\})$
    $R^\alpha(1) = P^{'}$ ; $R^\alpha(1) = \{\,\}$
    $U(\alpha) = V(\alpha) \setminus \{\rho\}$
    **while** $U(\alpha) \neq \emptyset$ **do**
      choice $\rho_i \in U(\alpha)$
      $U(\alpha) = U(\alpha) \setminus \{\rho\}$
      construct a chain decomposition $C(\alpha) = \{C_1^\alpha, C_2^\alpha, \ldots, C_q^\alpha\}$ of $R^\alpha$
    **end while**
    **for** i=1 **to** q **do**
      **I1:** do binary search on $C_i^\alpha$ using SD-QueryOracle to find smallest element (if any) that dominates $\rho_i$
      **I2:** do binary search on $C_i^\alpha$ using SD-QueryOracle to find largest element (if any) that dominates $\rho_i$
    **end for**
    infer all results $R^\alpha$ of **I1** and **I1** into $R^\alpha$:
      $R^\alpha(1) = R^\alpha(1) \cup \rho$
      $R^\alpha(2) = R^\alpha(2) \cup R^\alpha$
    find a chain decomposition $C^\alpha$ of $R^\alpha$
    construct ChainMerge($R^\alpha$, $C^\alpha$) data
  **end for**
  return ($V = [V_1, V_2, \ldots, V_r]$, ChainMerge(V)=[ChainMerge($V_i$),i=1,...,r])
**end function**

**Fig. 4.** Algorithm for partial ordered sorting entangled quantum states

and for qubits there exist exactly four such maximally entangled states. The one of the Bell states for qudits where $d = 3$ can be written similarly to the qubit case

$$|\psi\rangle = \alpha_0|00\rangle + \alpha_1|11\rangle + \alpha_2|22\rangle \quad \text{and} \quad |\alpha_0|^2 + |\alpha_1|^2 + |\alpha_2|^2 = 1 \ . \tag{6}$$

In general, the set of d-level Bell maximally entangled states for two qudits can be expressed through the following equation:

$$|\psi_{pq}^d\rangle = \frac{1}{\sqrt{d}} \sum_{j=0}^{d-1} \mathrm{e}^{2\pi\imath jp/d}|j\rangle|(j+q) \bmod d\rangle \ . \tag{7}$$

| def make_psi(r, p, q): | def make_psi(r, p, q): |
|---|---|
| r.Reset() | r.Reset() |
| for i in range(0,q): | for i in range(0,q): |
| r.NotN(1) | r.NotN(1) |
| r.HadN(0) | r.RandGateRealN(0) |
| for i in range(0,p): | for i in range(0,p): |
| r.PauliZ(0) | r.PauliZ(0) |
| r.CNot(0,1) | r.CNot(0,1) |

**Fig. 5.** The functions written in Python preparing the entangled Bell states for given register. The symbol denoted by $r$ is an object representing the quantum register and $p$ and $q$ are indices of Bell state generated by this procedure. The left column generates maximally entangled state but in the right column instead of the Hadamard gate we use a randomly generated gate to produce a non-maximally entangled states.

**Table 1.** The time consumed by sorting tests which use randomly generated quantum registers with a different amounts of entanglement in the sense of von Neumann's entropy

| Number of registers | Time (results in secs) |
|---|---|
| 10 | 0.0008762 |
| 100 | 0.0048304 |
| 1000 | 0.1407390 |
| 2000 | 0.4907356 |
| 4000 | 1.8053686 |
| 10000 | 10.643833 |

It is possible to express equation (7) in terms of qudit gates:

$$|\psi^d_{pq}\rangle = (I_d \otimes X_d)^q \cdot (H_d \otimes I_d) \cdot (Z_d \otimes I_d)^p \cdot \mathrm{CNOT}_d \cdot |00\rangle \quad . \tag{8}$$

where $0 \leq p \leq d-1$ and $0 \leq q \leq d-1$ are indices of one of $d^2$ allowed Bell state. The symbol $I$ represents the identity matrix for d-level qudit, and $H$ represents the Hadamard gate and $Z$ and $X$ are generalized Pauli's operators.

A simple function written in the Python programming language which uses the QCS module to generate entangled states is depicted in Fig. 5. We use this function to construct entangled states for earlier prepared quantum register.

The function presented in the left column of Fig. 5 generates states which have always the same amount of entanglement. Therefore the function from Fig. 5 must be equipped with some additional unitary gate to modify the entanglement amount. In the qubit cases the additional rotation gate after Hadamard gate can be used. In general any unitary gate that realises the rotation through any axis may be used to generate Bell states with uniform distribution of entanglement. Indeed, the right version of function *make_psi* from Fig. 5 possesses this feature.

Using function from Fig. 5 and the appropriate computational procedure to calculate the von Neumann entropy it is possible to prepare a simple benchmark. Additionally, to obtain comparable results we prepared simple test as a script in Python language for quantum registers built only from qubits. The test contains the following computation steps: first we generate n quantum registers then for every register the von Neumann entropy is calculated. After these steps we sort the obtained list using the classical method called sorting by selection. In Table 1 we present the real time necessary to perform this simple test.

### 2.4 Computational Complexity Analysis

The computational complexity of algorithm in Fig. 1 is given by following equation:

$$T(n) = \sum_{i=1}^{N} (T_1(n_i, d_i) + T_2(n_i, d_i) + T_3(n_i, d_i)) + T_{sort}(n) \tag{9}$$

where $N$ is the total number of quantum registers and $d_i$ represents freedom level of qudit used in given $n_i$ quantum register. Additionally, $T_1(\cdot)$ represents the complexity of partial trace calculation, $T_2(\cdot)$ is the complexity of calculation of the eigenvalues and eigenvectors and $T_3(\cdot)$ is the complexity of the von Neumann entropy calculation. Each of mentioned complexity functions work on matrices and if we assume that $n$ is the size of matrix and $d$ is freedom level of a state which is given by density matrices we obtain the following relations:

$$T_1(n, d) = dn^2, \quad T_2(n, d) = n^3, \quad T_3(n, d) = n \ . \tag{10}$$

The complexity of $T_{sort}(n)$ depends on the algorithm used to sort the obtained quantum registers, the value of entropy is used to compare two registers. If we use one of the popular sorting methods like Heapsort with complexity given by $O(n \log(n))$, the complexity of algorithm in Fig. 1 will be

$$T(n) = N(dn^2 + n^3 + n) + n \log(n) \ , \tag{11}$$

where the process of computation of eigenvalues and eigenvectors is the most time-consuming part of the whole process of sorting quantum registers.

The second algorithm of sorting quantum states (Fig. 4) contains oracle routine as described by algorithm in Fig. 3. The complexity of oracle for the worst case when the ranks are equal is given by:

$$T(n) = n^3 + n \log(n) + n^2 = O(n^3) \ , \tag{12}$$

The procedure of calculation the singular value decomposition dominates the computational complexity of oracle routine. It is important to stress that in the oracle routine we also sort the Schmidt coefficients, but by using the classically effective algorithm. However, the SVD still dominates the complexity of SD-QueryOracle.

It is known [7] algorithm in Fig. (4) calls the query at most $\mathcal{O}(w \cdot n \log n)$, where $w$ is the maximal width of poset containing $n$ elements but the time of SVD again dominates the whole process of partial sorting of quantum states.

## 3 Conclusions and Further Work

Basing on the canonical Schmidt decomposition of quantum states a specific semiorder relation has been introduced in the space of quantum states of a given bipartite system. In the case of pure states the introduced semiorder relation possesses a very clear operational meaning as described by Nielsen [4] for the first time. Whether the same operational meaning can be affiliated with the analogous semiorder relation defined in the space of all quantum states should be explained.

Additionally, some version of sorting algorithm of the arising posets, the so called ChainMerge sorting and basing on the particular version of query oracle comparing the amount of entanglement in two quantum states is presented and tested in the case of vector states. The following extensions of the present material seems to be worthwhile to perform: (a) to extend the Nielsen result [4] to cover the case of general quantum states, (b) to formulate several different version of sorting posets algorithm with special emphasis putted on their computational complexity, (c) to formulate different version of query oracles for comparing the amount of entanglement included in two general states of bipartite systems.

## References

1. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information. Cambridge University Press, Cambridge (2000)
2. Garey, M.R., Johnson, D.S.: Computers and Intractability: A guide to the Theory of NP-Completeness. W.H.Freeman and Co., San Francisco (1979)
3. Papadimitrio, C.H.: Computational Complexity. Addison-Wesley, Reading (1994)
4. Nielsen, M.A.: Conditions for a class of entanglement transformations. Phys. Rev. Lett. 83, 436 (1999)
5. Gurvits, L.: Classical deterministic complexity of Edmonds – problem and Quantum Entanglement. In: Proc. of the 35th Annual ACM Symposium on Theory of Computing, San Diego, USA, June 9-11 (2003), also available at quant-ph/0303055
6. Daskalakis, C., Karp, R.M., Mossel, E., Riesenfeld, S., Verbin, E.: Sorting and Selectio In Posets, http://arxiv.org/abs/0707.1532
7. Faigle, U., Turan, G.: Sorting and Recognition Problems for ordered Sets. SIAM J. Comput. 17(1), 100–113 (1988)
8. Gielerak, R., Sawerwain, M.: Sorting of amount of Entanglement in Quantum States functions implemented for Quantum Computing Simulator. Submitted to KNWS 2009 conference, www.knws.uz.zgora.pl
9. Terhall, B.M., Horodecki, P.: arXiv:quant-ph/9911117v4
10. Chen, K., Wu, L.A.: Quant. Inf. Comp. 3, 193 (2003)
11. Rudolph, O.: Phys. Rev. A 67, 032312 (2003)
12. Gielerak, R., Sawerwain, M. (in preparations)
13. Sawerwain, M., Gielerak, R.: Natural quantum operational semantics with predicates. Int. J. Appl. Math. Comput. Sci. 18(3), 341–359 (2008)

# Management of Web Services Based on the Bid Strategy Using the User Valuation Function*

Jolanta Wrzuszczak and Leszek Borzemski

Institute of Computer Science, Wroclaw University of Technology Wroclaw, Poland
{jolanta.wrzuszczak,leszek.borzemski}@pwr.wroc.pl

**Abstract.** In the chapter the auction methods based on the bidding approach will be proposed to be employed both in scheduling and controlling of Web services. The aim of the underlying Web service will be the management of user requests, which will revenue their valuation function to get server resources based on the fuzzy-logic approach. Two strategies will be analyzed and justified through experiments made in the Matlab environment. The first situation deals with the resources being in deficiency to serve all clients, whereas in the second one, every user will be served on the expected level and some resources will be left not assigned, so that the volume will be assigned to as "additional" goods for other users. In this work a scheduling performance based both on the auction approach and on a FIFO policy will be evaluated and compared. It will be shown how efficient the bidding algorithm could be in the real life applications.

## 1 Introduction

Making money transactions via Internet is becoming more and more popular. Many services require paying charges for their resources. To implement such approaches, efficient scheduling algorithms must be applied and a proper QoS must be guaranteed. Additional strategies must take place, to assure the desired level of user's satisfaction. In the literature many works appeared dealing with the problem of managing the service resources [3,4,5,6,8].

Our aim is to study the possibility of providing QoS service based on auction approaches in which a price paid by users for a service isn't strictly defined. Then the usage of an auction algorithm combined with aspects of fuzzy logic methodology may be applied [7]. In this work the service management approach, applying an auction approach based on fuzzy logic, will be developed and discussed. Effectiveness of proposed strategies will be evaluated in the experiments running in the Matlab environment.

The idea of reporting offers (requests) for service resources will be extensively discussed in the next section. The third section presents a concept of implementing the auction method based on the fuzzy logic approach in a Web service. In

* This work was supported by the Polish Ministry of Science and Higher Education under Grant No. N516 032 31/3359 (2006 2009).

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 19–25, 2009.

the fourth section the problem of scheduling jobs in a Web service will be formulated. Also, some analysis comparing the original proposed approach versus a FIFO policy will be discussed based on simulated runs, performed in the Matlab environment. In the summary some concluding remarks will be given.

## 2 User Valuation Function

The role of a proposed Web service will provide a suitable bandwidth for downloading files. Pieces of bandwidth will be the auctioned items. Users will be required to reveal their valuation function based on the R-(membership) distributive function [1], so they will report the initial requests (a minimal bandwidth they are interested in) and the upper bound for which they will offer the highest price [2]. After reaching the point of maximal level, the price will remain constant.

Every user is obliged to reveal his valuation function $\theta_i$ fulfilling the following requirements:

1. $\theta_i(0) = 0$ at the starting point the value of the valuation function equals zero [9],
2. $\theta_i$ is non-decreasing function [9],
3. $\theta_i$ is a continuous, derivable and linear function at the first stage and then remains constant (saturated),
4. The maximum value of the valuation function is 1 [7].

In Fig. 1 three exemplary valuation functions are presented.

After the bidding process (collecting offers) the server will know each valuation function. The aim will be to manage the bandwidth. Two strategies may be considered. In the first strategy the total volume of bandwidth is smaller than the total sum of expected bandwidth reported by all clients. In the second strategy, the total sum of user requests will be smaller than the overall bandwidth

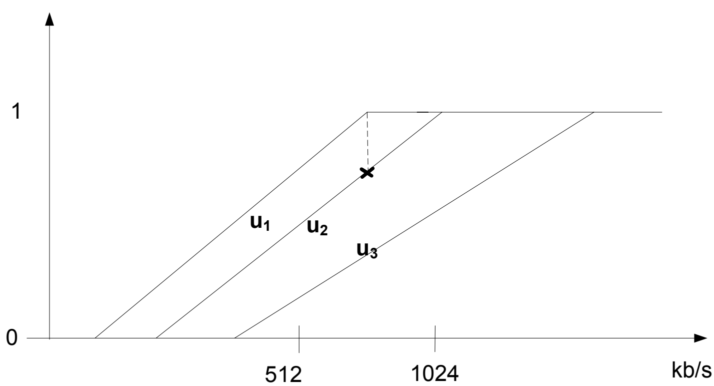



**Fig. 1.** Valuation function reported by three users

proposed by the server. Thus, the remaining bandwidth would be distributed among the users. The proposed mechanisms will be treated in details in Sect. 3.

## 3 Algorithms and Problem Formulation

The aim of Web service will be to support the possibility of delivering multimedia files, for which users will be obligated to pay a charge related to a demanded bandwidth. Proposed mechanisms will be considered among clients and the scheduler of Web service (Fig. 2). Two cases will be discussed.

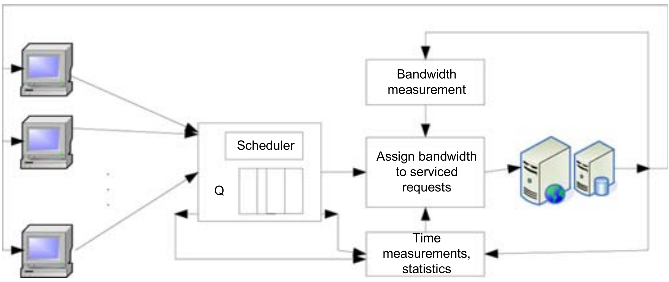



**Fig. 2.** Schema of web service

### 3.1 First Case

Server's resources aren't sufficient to satisfy all clients. The total bandwidth is limited, so resources will be granted only to some number of clients. Every user that is served will receive requested bandwidth. Three algorithms: FIFO, First Price and Vickrey will be considered in providing this policy.

**FIFO Algorithm.** From queue of submitted valuation functions, clients with the earliest time stamp of reported offers will be served. The offers will be chosen for such a long time that the server exhausts the total bandwidth. The income of the Web service ($\varphi_1$) will be calculated as a sum of charges paid by the users (Eq. 1). The price paid by clients will be determined by their declared charge.

$$\varphi_1 = \sum_{i}^{n} b_i \tag{1}$$

**First Price Auction Algorithm.** This mechanism supports who offers the top price ("1") for minimal piece of bandwidth. Resources will be distributed to clients in the sequence according to the price they have offered [11].

The income of the Web service ($\varphi_2$) will be calculated as a sum of charges paid by the users (Eq. 2).

$$\varphi_2 = \sum_{i}^{m} c_i \tag{2}$$

**Auction Algorithm Based on Vickrey Approach.** In this approach, the clients will be chosen as in the first price auction policy. The difference is that the price paid by users will be determined by other users' offers [2,10,11]. For defined bandwidth the highest price reported by the other client is determined. The value is calculated from other users' valuation functions using the bandwidth argument, i.e. the X in Fig. 1 determines the price for the first user ($u_1$).

In the Vickrey approach for every offer the price will be calculated based on the second highest price offered in the current bid. The income of Web service ($\varphi_3$) will be defined as a sum of served offers (3).

$$\varphi_3 = \sum_{i}^{k} d_i(p), d_i(p) = \max_{j \neq i} f_j(p) \tag{3}$$

where ($f_j(p)$)valuation function for user ($j$) and number of bandwidth pieces ($p$).

### 3.2 Second Case

The total sum of user requests doesn't exceed the total volume of server resources. The remaining bandwidth could be assigned to some users.

FIFO and First Price algorithms will exhibit the same results because every client will be served and will pay his declared price. Therefore in the second case only two algorithms will be considered FIFO/First Price and Vickrey.

**FIFO/First Price.** In this approach the remaining bandwidth will be assigned for free, so any additional income of the Web service will be supported. The performance index ($\varphi_4$) (Eq. 4) will be defined as a total sum of bandwidth pieces which will be sold to $u$ users.

$$\varphi_4 = \sum_{i}^{u} l_i \tag{4}$$

**Auction Algorithm Based on the Vickrey Approach.** The remaining bandwidth will be assigned in the same manner as in a Vickrey algorithm presented in the first case but the price will be changed. The supplement of bandwidth will be assigned as 30% more than expected and in this case the price will be determined by a reported valuation function.

The performance index ($\varphi_5$) (Eq. 5) will be defined as a total sum of bandwidth pieces, which will be sold to $u$ users.

$$\varphi_5 = \sum_{i}^{u} r_i \tag{5}$$

In the Vickrey algorithms, assigning the base volume to clients will bring lower income of the Web service comparing to FIFO/First Price approach, because every client pays the second highest price for the assigned bandwidth.

**Problem Formulation.** For given:

- set of user valuation functions (distributive R-function),
- performance index,

should be found a strategy, which optimizes the performance index and assures the client satisfaction simultaneously.

**Experiment.** The Vickrey algorithm for unique items, First Price and FIFO policy will be proposed to solve this problem. To evaluate discussed mechanisms some simulations have been performed.

The simulations have been conducted in the Matlab environment ver. 6.5 and Statistics Toolbox Version 4.0. The stream of offers (valuation function) was simulated as a stochastic process with a Poisson distributive function with the following values of $\lambda$ parameter ($\lambda$=500, 400, 200, 100, 75, 50, 25, 10, 5) . The parameter is the mean of total bandwidth reported by users. The number of users has been simulated as 1000 and 100. In the first case (bandwidth assigned only to chosen users) the bandwidth pieces were established at the level of 3000 and 6000, respectively.

The income of the Web service has been measured with performance indexes defined in Eq. 1, 2, 3 for the first approach. The total sum of bandwidth pieces was measured with performance indexes Eq. 4 and 5 for the second approach. The results of the first approach are showed in Fig. 3. Each pattern of the bar represents a total sum of charges for a different algorithm.

The second case was examind from the perspective of the volume of sold bandwidth pieces for a defined price. Every user will be served on the expected level, so they will receive a requested bandwidth. The total sum paid by the users in FIFO/First Price Policy will be the same and every additional bandwidth will be assigned for free. The server income will be calculated for the requested bandwidth pieces. After assigning some undistributed bandwidth pieces, the income of the Web service will be constant and the provided bandwidth pieces will arise.

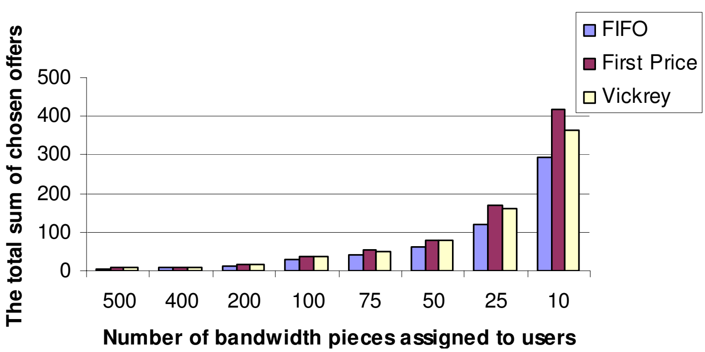



**Fig. 3.** The total sum of chosen offers versus the number of bandwidth pieces assigned to users for FIFO, First Price and Vickrey algorithms (for the first case)

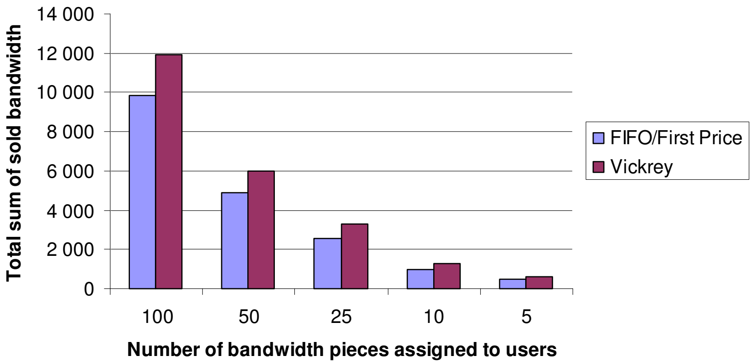



**Fig. 4.** The total sum of sold bandwidth versus the number of bandwidth pieces assigned to users for two algorithms (for second approach)

In the Vickrey approach the price is calculated for "requested" bandwidth pieces and will be lower in comparison to FIFO/First Price mechanism, but the server will earn "extra" resources for sharing them in the final stage. In Fig. 4 the total bandwidth versus a defined price is presented. The difference between bars confirms that the Web server income decreases as the server has divided the additional bandwidth for free at FIFO/First Price algorithm.

## 4 Results and Conclusions

Analyzing the performance index for three algorithms (FIFO, First Price and Vickrey for first discussed approach) presented in Fig. 3 shows, that in every case the best solution is the First Price auction algorithm, whereas the FIFO algorithm is the worst one. The Vickrey policy guarantees the client satisfaction because served clients will pay lower price than they declared and the value of performance index for Vickrey policy was a little bit worse when compared to the First Price algorithm.

In Fig. 4 the values of the performance index (sold bandwidth for the second case) have been shown. Every client of the Web service will be served on the expected level and the additional, unused bandwidth will be split for clients in FIFO/First Price policy for free. In Vickrey policy some additional bandwidth will be sold. In FIFO/First Price algorithms the sold bandwidth was lower when compared to Vickrey approach.

Summarizing the first approach, the Vickrey algorithm could be judged as a very effective because the value of its performance index was higher than in FIFO approach and a little bit worse than for a First Price policy. In the second approach, the algorithm based on Vickrey mechanism could be treated as very effective for managing server resources because more bandwidth was sold than for FIFO/First Price policy.

## References

1. Bobillo, F., Straccia, U.: An expressive fuzzy description logic reasoner. Fuzzy Systems IEEE, 923–930 (2008)
2. Borzemski, L., Wrzuszczak, J., Kotowski, G.: Management of Web service delivering multimedia files based on the bid strategy. Information Systems Architecture and Technology ISAT, 13–23 (2008)
3. Brazier, F., Cornelissen, F., Gustavsson, R., Jonker, C., Lindeberg, O., Polak, B., Treur, B.: A multi-agent system performing one-to-many negotiation for load balancing of electricity use. Electronic Commerce Research and Applications 1, 208–222 (2002)
4. Cherkasova, L., Phaal, P.: Peak Load Management for commercial Web servers using adaptive session-based admission control. In: Proceedings of the 34th Hawaii International Conference on System Sciences (2001)
5. Courcoubetis, C., Dramitinos, M., Stamoulis, G.: Auction-based Resource Allocation in UMTS High Speed Downlink Packet Access (HSDPA), pp. 434–441. IEEE, Los Alamitos (2005)
6. Cramton, P., Shoham, Y., Steinberg, R.: Combinatorial auctions. The MIT Press, Cambridge (2006)
7. Goyal, M., Lu, J., Zhang, G.: Decision Making in Multi-Issue e-Market Auction Using Fuzzy Techniques and Negotiable Attitudes. Journal of Theoretical and Applied Electronic Commerce Research 3, 97–110 (2008)
8. Maille, P., Tuffin, B.: Pricing the Internet with multibid auctions. IEEE/ACM Trans. on Networking 14(5) (2006)
9. Perez-Bellido, A., Salcedo-Sanz, S., Portilla-Figueras, J.A., Ortiz-Garcia, E.G., Garcia-Diaz, P.: An Agent System for Bandwidth Allocation in Reservation-Based Networks using Evolutionary Computing and Vickrey Auctions. In: Nguyen, N.T., Grzech, A., Howlett, R.J., Jain, L.C. (eds.) KES-AMSTA 2007. LNCS (LNAI), vol. 4496, pp. 476–485. Springer, Heidelberg (2007)
10. Vickrey, W.: Counterspeculation, Auctions, and Competitive Sealed Tenders. The Journal of Finance 16(1), 8–37 (1961)
11. Wrzuszczak, J.: Auction mechanism in management of processing nodes in a computer cluster. Contemporary Aspects of Computer Networks 2, 259–265 (2008)

# A Markovian Model of a Call Center with Time Varying Arrival Rate and Skill Based Routing*

Jarosław Bylina[1], Beata Bylina[1], Andrzej Zoła[2], and Tomasz Skaraczyński[2]

[1] Institute of Mathematics
Marie Curie-Sklodowska University
Pl. M. Curie-Skłodowskiej 1, 20-031 Lublin, Poland
beatas@hektor.umcs.lublin.pl
jmbylina@hektor.umcs.lublin.pl

[2] Institute of Mathematics and Computer Science
John Paul II Catholic University of Lublin
ul. Konstantynów 1H, 20-708 Lublin, Poland
andyz@kul.pl
skarat@kul.pl

**Abstract.** The paper aims at construction and investigation of a practical model serving a call center performance evaluation. The system considered here consists of two queues and three agents' classes. For such a system we develop a queuing model with the arrival (calling) rate varying with time and a continuous-time Markov chain (CTMC) which provide approximations of system performance measure. The results obtained from the CTMC are compared to results obtained from a simulation of a call-center. The models are applied to determine: average time of answer, probabilities of waiting in queues, average length of queues.

## 1 Introduction

The problem of a call center functioning is a very important issue from the point of view of various disciplines: economy, teleinformatics, sociology, psychology. The subject of the teleinformatics is – among many others – to forecast the performance of call centers with the use of simulation [7,8,9,10] and analytical modelling [1,4,6].

Both manners – simulation and analytical modelling – have their own advantages and flaws.

The simulation is an easy subject to modifications and can be simply adapted to every system. Moreover, the arbitrary accuracy can be achieved by simulation – given sufficient time. However, simulation can take a lot of time, especially when precise results are needed.

On the other hand some analytical methods can be used. They consists in making a mathematical model of the investigated system and then solving some

* This work was partially supported within the project *Metody i modele dla kontroli zatloczenia i oceny efektywnosci mechanizmow jakosci uslug w Internecie nastepnej generacji* (N517 025 31/2997).

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 26–33, 2009.

equations describing such a model. From their solutions various characteristics of the system can be obtained.

Among analytical methods we can name: discrete approximation, diffusion approximation, mean value analysis, stochastic Petri nets and many others. But as the most intuitive and natural way of modelling call centers we chose Markovian queuing models solved with the use of continuous time Markov chains (CTMCs). Such analytical models can give better accuracy in shorter time than simulation mentioned above.

The simplest queuing model describing a very simple call center is $M/M/s$ [4,6] describing a system with clients of only one class, with $s$ identical serving agents, and with exponential arriving time and exponential serving time.

Some more complicated Markovian models for call centers were considered in [3,5,13].

The authors in [11] presented a queuing model with clients of two classes and three (the case with skill based routing – also written SBR – which means that incoming calls are assigned to the most suitable agent and some agents are proficient in serving both clients' classes) or two (the case without SBR) agents' classes. The model was solved with analytical methods and with simulation. There was an assumption that the arrival and serving rate is constant in time and only steady states probabilities were investigated.

The present paper is a natural extension of those studies. Here, the authors try to find transient probabilities of various events and, moreover, for variable arrival rate. Such an approach certainly approximates a real call center better, where the number of connections varies in time.

The results are achieved in two ways – with simulation and with numerical solving of some differential equations with the fourth-order Runge-Kutta method [12].

This paper is organized as follows. Sect. 2 explain how we construct our CTMC model where the rate of clients' arrival depends on time. In Sect. 3 we discuss some details of the simulation model. In Sect. 4 we compare the performance of the CTMC models with the simulation results for an example of a real call center. We also briefly outline how the CTMC model might be used for optimal scheduling. We conclude in Sect. 5 with a summary and future research directions.

## 2 CTMC Model

Consider a scheme of a call center presented in Fig. 1. The system consists of two queues and three agents' classes. Let us denote:

- $q_{\mathrm{T}}$, $q_{\mathrm{B}}$ – the maximum number of clients in the queues $Q_{\mathrm{T}}$ and $Q_{\mathrm{B}}$, respectively;
- $s_{\mathrm{T}}$, $s_{\mathrm{B}}$, $s_{\mathrm{TB}}$ – the number of agents in groups $S_{\mathrm{T}}$, $S_{\mathrm{B}}$ and $S_{\mathrm{TB}}$, respectively;

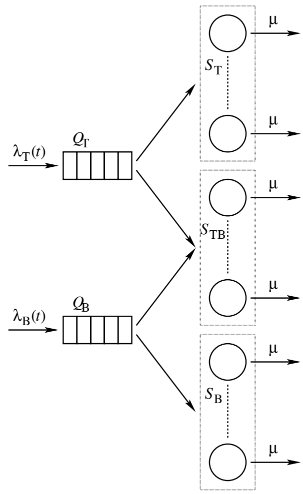



**Fig. 1.** A queuing model for the investigated call center

- $\lambda_T(t)$, $\lambda_B(t)$ – the intensity of incoming traffic (i.e. the mean value of arrival time) of clients of class T and B, respectively;
- $\mu$ – the intensity of outgoing traffic (i.e. the mean value of service time).

We will define the state of the system in the following way:

$$X(t) = (n_T(t), n_B(t), n_{TB}(t)) \tag{1}$$

where $n_T(t)$ denotes the quantity of clients served by agents from the group $S_T$ together with the number of clients in the queue $Q_T$ – at the moment $t$. Similarly, $n_B(t)$ denotes the quantity of clients served by agents from the group $S_B$ together with the number of clients in the queue $Q_B$ – at the moment $t$. Finally, $n_{TB}(t)$ denotes the quantity of clients served by agents from the group $S_{TB}$ – at the moment $t$.

The modelled call center is one with skill based routing, which is realized as follows. As long as there are free agents in the group $S_T$, clients from the queue $Q_T$ are directed to them – as is the case of $S_B$ and $Q_B$, respectively. When there are clients in $Q_T$ and there are no free agents in $S_T$, the clients are directed to agents from $S_{TB}$; similarly for $Q_B$ and $S_B$.

So the constraints for the elements of the system state $(n_T(t), n_B(t), n_{TB}(t))$ are (we omit $t$ for clarity):

$$\begin{cases} n_{TB} \in [0, s_{TB} - 1] \\ n_T \ \ \in [0, s_T] \\ n_B \ \ \in [0, s_B] \end{cases} \quad \text{or} \quad \begin{cases} n_{TB} = s_{TB} \\ n_T \ \ \in [0, s_T + q_T] \\ n_B \ \ \in [0, s_B + q_B] \end{cases} . \tag{2}$$

Every possible transitions of the system from any given state $(n_T, n_B, n_{TB})$ are presented in Fig. 2.

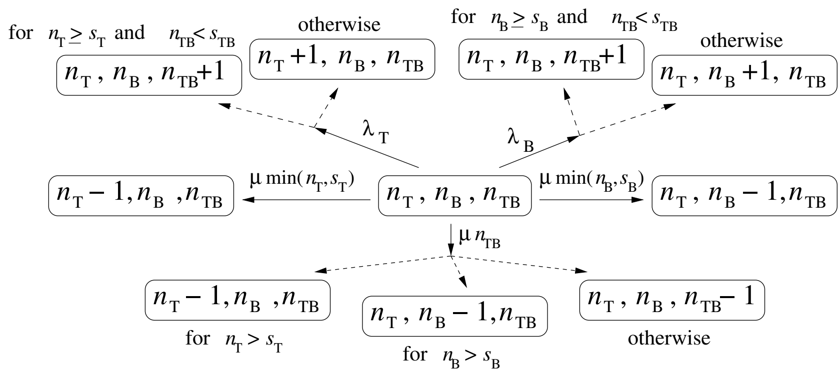



**Fig. 2.** Possible transitions among states in the presented model

To find transient probabilities $\boldsymbol{\pi}(t)$ of states – and then some characteristics of the model – we were to generate a transition rate matrix $\mathbf{Q}(t)$ for our CTMC and then solve a differential equation

$$\frac{d\boldsymbol{\pi}(t)}{dt} = \boldsymbol{\pi}(t)\mathbf{Q}(t) \ . \tag{3}$$

## 3 Simulation Model

The CQMS (Component Queue Modelling System) [9,10] simulation platform was developed by the authors of this article as a system aiding building network models and testing them in simulation. The main goal was to build a tool that enables creating models of various network easily. To achieve this aim a component approach was used, which provided high flexibility. The system consists of components representing elements of network queue models, such as clients, groups of agents, etc. The software also includes some visual components that allow to present the results of simulation as well as to visualize the network in work. Building a network model in CQMS requires using appropriate components and then linking them by means of properties and events.

## 4 Tests, Results and Comparison

In tests, some parameters were fixed as follows:

- $q_T = 10$, $q_B = 10$;
- $s_T = 10$, $s_B = 15$, $s_{TB} = 0$ for a non-SBR system and $s_T = 8$, $s_B = 12$, $s_{TB} = 5$ for an SBR system;
- $\mu = 0.0022$ for a roughly balanced system and $\mu = 0.002$ for a slightly overloaded system.

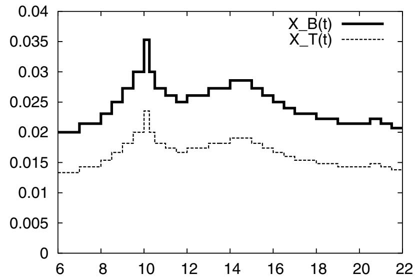
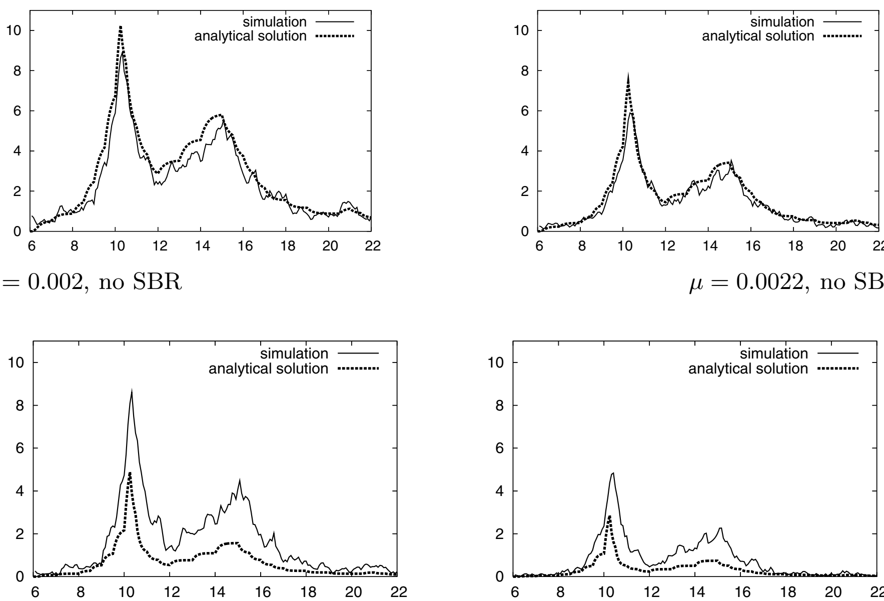



**Fig. 3.** Arrival rates $\lambda_{\mathrm{T}}(t)$ and $\lambda_{\mathrm{B}}(t)$ as functions of time (time given in call center work-hours: 6 am till 10 pm)




$\mu = 0.002$, no SBR

$\mu = 0.0022$, no SBR

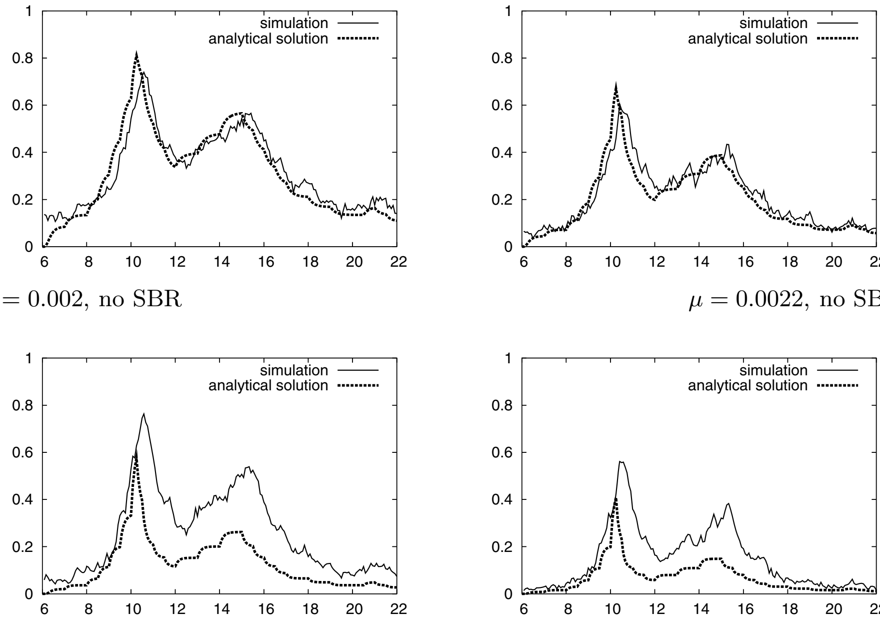
$\mu = 0.002$, with SBR

$\mu = 0.0022$, with SBR

**Fig. 4.** Average total length of queues

Arrival rates $\lambda_{\mathrm{T}}(t)$ and $\lambda_{\mathrm{B}}(t)$, being time-dependent, are shown in Fig. 3. All rates ($\mu$, $\lambda_{\mathrm{T}}(t)$ and $\lambda_{\mathrm{B}}(t)$) are given in $[\mathrm{s}^{-1}]$ (that is, per second). These values are based on data from a real call center originated from [4].

For a numerical solution of the model (see also Sect. 2) we generated the transition rate matrix $\mathbf{Q}(t)$ on the basis of the transition graph (Fig. 2) numbering possible states in the lexicographical order. The matrix was not very big (1022 states with SBR, 546 states without SBR), so we did not employ any parallel or distributed algorithm for its generation (as we employed in [2]). The equation




$\mu = 0.002$, no SBR

$\mu = 0.0022$, no SBR

$\mu = 0.002$, with SBR

$\mu = 0.0022$, with SBR

**Fig. 5.** Probability that the client has to wait

(3) was also not very demanding and was solved with the use of fourth-order Runge-Kutta method with the step $h = 0.01\,[\mathrm{s}]$.

The simulation was conducted with time step equal to $1\,[\mathrm{s}]$, although the results were aggregated every 5 minutes during each workday (i.e. a mean was calculated for every 300 steps). All the tests were repeated 1000 times.

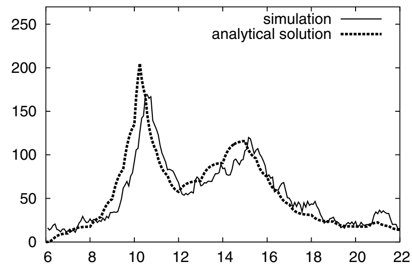
The following characteristics were obtained (among others) and analyzed:

– average total length of queues (for a call center it denotes the number of clients connected to the center's telephone network but not currently being served) – Fig. 4;
– probability that an arriving client has to wait for an agent's answer (regardless of the time of waiting) – Fig. 5;
– average time of an agent's answer (the average time that a client spends in a queue, i.e. "on hold") – Fig. 6.

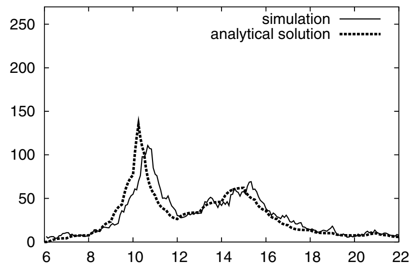
All the graphs in these figures are given with the time of call center work-hours: 6 am till 10 pm. Each measure is given for four different models differing with $\mu$ and existence of SBR (as was declared at the beginning of this section).

It can be seen in the figures that both methods give expected results. That is:

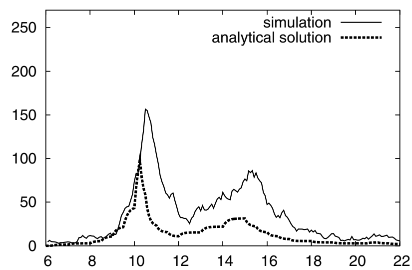
– the analytical solution gives results similar to the simulation (although differences are more significant for models with SBR; on the other hand, the resemblance for models without SBR is very strong);



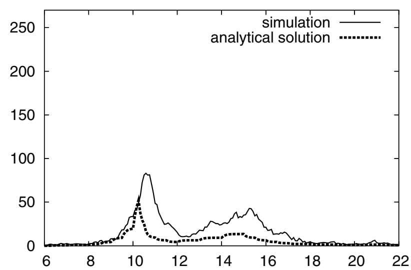

$\mu = 0.002$, no SBR




$\mu = 0.0022$, no SBR




$\mu = 0.002$, with SBR




$\mu = 0.0022$, with SBR

**Fig. 6.** Average time of answer

- the characteristics obtained for bigger $\mu$ are better (that is smaller) the for smaller $\mu$;
- the same can be seen for SBR – better characteristics with SBR, as it can be expected;
- when $\lambda_{\mathrm{T}}(t)$ and $\lambda_{\mathrm{B}}(t)$ grow, then the studied characteristics also grow (that is get worse) – what is mainly visible about 10 am.

The difference between the simulation and the analytical solution for cases with SBR requires an individual explanation. It is a well known trait of simulation methods that they demand a lot of repetitions to acquire reliable results – especially when we deal with a lot of different states which are not very probable (each on its own), but together can play a significant role. Here we have more states for models with SBR than without it (1022 to 546) so some rare (but not unimportant) events are more likely to be lost in the simulation.

## 5 Conclusion

We have studied a Markovian model with time varying arrival rate. We compared estimated performance measures obtained from this model with measures obtained from a detailed simulation model of a call center.

Both analytical and simulation results show clearly that skill based routing improves the work of a call center. The improvement is most significant in cases of normal load, i.e. when the number of clients calling is close but lower than the number of clients that the center is able to serve. It can also be seen that the call center performance in rush hours is quite sensitive to little changes of $\mu$ and (as expected) to the presence of SBR.

## References

1. Bhulai, S., Koole, G.: A queueing model for call blending in call center. IEEE Trans. on Aut. Contr. 48, 1434–1438 (2003)
2. Bylina, J., Bylina, B.: Development of a distributed algorithm of matrix generation for Markovian models of congestion control and its performance analysis on a computer cluster. In: Contemporary Aspects of Computer Networks, Wydawnictwa Komunikacji i Łaczności, Warszawa, vol. 2, pp. 251–258 (2008)
3. Deslauriers, A., L'Ecuyer, P., Pichitlamken, J., Ingolfsson, A., Avramidis, A.N.: Markov chain models of a telephone call center with call blending. Computers & OR 34(6), 1616–1645 (2007)
4. Gans, N., Koole, G., Mandelbaum, A.: Telephone Call Centers: tutorial. Review and Research Prospects, Manufact. and Service Oper. Manag. 5, 79–141 (2003)
5. Ingolfsson, A.: Modeling the M(t)/M/s(t) Queue with an Exhaustive Discipline (submitted)
6. Koole, G., Amndelbaum, A.: Queueing Models of Call Centers. An Introduction. Ann. of Oper. Res. 113, 41–55 (2002)
7. Perrone, L.F., Wieland, F.P., Liu, J., Lawson, B.G., Nicol, D.M., Fujimoto, R.M. (eds.): Variance Reduction in the Simulation of Call Centers. In: Proceedings of the 2006 Winter Simulation Conference (2006)
8. Pichitlamken, J., Deslauriers, A., L'Ecuyer, P., Avramidis, A.N.: Modelling and simulation of a telephone call center. In: Proceedings of the 2003 Winter Simulation Conference, pp. 1805–1812 (2003)
9. Skaraczyński, T., Zoła, A.: Komponentowy system modelowania kolejek w zastosowaniu do telefonicznych centrów obsługi. Współczesne aspekty sieci komputerowych, Wydawnictwa Komunikacji i Łaczności, Warszawa, vol. 1, pp. 89–96 (2008) (in Polish)
10. Skaraczyński, T., Zoła, A.: Optimizing employment in telephone call centers. Theor. and Appl. Inform. 20(1), 39–47 (2008)
11. Skaraczyński, T., Zoła, A., Bylina, J., Bylina, B.: Markovian method of modeling call-centers. In: Computer Science and Information Technologies (Proceedings of CSIT 2008), Lviv, pp. 51–54 (2008)
12. Stewart, W.J.: An Introduction to the Numerical Solution of Markov chains. Princeton University Press, New York (1994)
13. Whitt, W.: Engineering Solution of a Basic Call-Center Model. Manag. Sci. 51(2), 221–235 (2005)

# A Novel Multicast Routing Protocol for Mobile Sensor Networks with Topology Control

Jerzy Martyna and Marcin Nowrot

Institute of Computer Science, Jagiellonian University, ul. Lojasiewicza 6, 30-348 Cracow, Poland
{martyna,nowrot}@softlab.ii.uj.edu.pl

**Abstract.** In this paper, a novel protocol, called *Multicast Routing Protocol for Mobile Sensor Networks* (MuPMS), is proposed. It is based on the two-phase creation of multicast trees in a mobile network. During the movement our protocol can modify all previously built multicast trees. It allows for the better data harvesting by a mobile node. Both analysis and simulation results show that the performance of given protocol is fair and can be significantly tuned according to application requirements.

## 1 Introduction

*Wireless Sensor Networks* (WSNs) [1] are often used for environment monitoring. The applications implemented in these networks must be able to find defined regions, gather data and send it to special nodes which are referred to as the sink(s). Sinks are used as gateways to the Internet or host computers. If sinks or certain nodes in a WSN are mobile, then the WSN is defined a *Mobile Sensor Network* (MSN).

In general, certain nodes (or a single node) in MSN can be moved by using their own control system or natural motion, such as sea currents, a strong wind, etc. It is possible that an MSN can translocate according to the mobile model. Thus, mobile nodes have the ability to pattern recognition, communication and control, etc. The main difference between MSNs and static WSNs lies in data gathering and transmitting processes. In a static WSN all queries or messages are sent in the forecast way. In the case of MSNs it depends on the mobile model of the nodes in the system.

The concept of the MSN and its activities are described in a number of papers [2,3,4,5]. Localizations in MSNs are given in research works [6,7,8]. Examples of MSN are given in papers [9,10].

The multicast tree problem in the context of delivering data or multimedia streams in computer network consists in building routing trees from a source to a several destinations which are devoted to data transfer. The resulting data transfer has substantially lower bandwidth costs compared to the two alternatives, i.e. sequential unicast transmission or flooding.

The multicast process in mobile networks differs from static networks because the nodes may use omnidirectional transmissions. During a transmission

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 34–49, 2009.

all neighbouring nodes receive the same information in a single transmission. Therefore, the goal of multicast tree building is defined as the process of minimizing the number of transmissions that take place.

Multicast protocols in mobile ad hoc networks has been described in a number of papers [11,12,13,14]. A paper by Chandra [12] proposed the improving multicast reliability. A route driven gossip protocol was introduced by Luo [13]. A congestion-controlled multicast protocol was given by Tang [14]. An overview of multicast protocols which can be implemented in mobile ad hoc networks is outlined in a paper by Obraczka [15].

The main goal of this paper is to provide a new distributed protocol which can be implemented in mobile sensor networks. This allows us to construct multicast trees for gathering data and next to send the data to moving nodes. In contrast to all well-known multicast protocols used in mobile ad hoc networks, our protocol is employed in a mechanism for topology control. In other words, it can define for all nodes the transmitting powers, which are not greater than is necesarry.

The rest of the paper is organized as follows. Section 2 presents the multicast routing protocol for WSNs with topology control, which leads to a minimization of energy consumption. We then describe in detail the activity of our protocol. Performance analysis and an energy-efficiency analysis of our protocol are made in Sect. 3. Simulation results are given in Sect. 4. Finally, the last section concludes the paper.

## 2 Multicast Routing Protocol for Mobile Sensor Networks

WSNs and ad hoc networks use three categories of multicast protocols: Automatic Retransmission Request (ARQ) based protocols [16], gossip-based protocols [17] and protocols based on the FEC (Forward Error Correction) mechanism [18]. The first category includes those multicast protocols in which all packets are retransmitted until they are received by all the receivers. In gossip-based protocols, multicast packets are repeatedly transmitted a few times by a few multicast members in peer-to-peer fashion. Multicast protocols based on a FEC mechanism embed redundant data (e.g. erasure code) in each packet before transmitting. A few packet losses are tolerated in these protocols and the original data can be reconstructed by using correctly received ones.

Let a MSN be a set of $n$ nodes. All nodes can communicate with each other via wireless links, when they are in transmitting range. Each node has in its transmitting range a restricted number of sensor nodes. MSN nodes and possible wireless links are described in directed graph $G(V, E)$, where $V$ is a set of all nodes and $E$ is a set of all edges which are directed wireless links. We assume there exists here a set $\tau_i$, $\tau_i \subseteq V$, which only contains roots of multicast trees. It is supposed that the MSN works in a synchronous node in which the time axis is divided into identical time intervals, called slots. The length of a slot is equal to the duration of the transmission frame. During the time transmission of a frame each node can send or receive only one frame. The synchronous mode of data transmission respond to the TDMA/FDMA access method.

The activity of the multicast routing protocol for an MSN involves two phases here. In the first phase during the movement of a mobile node are sent periodically packets which initiate multicast tree building. Each node after receiving this packet as the first from among others sends in the broadcast mode a packet announcing the initiation of a multicast tree. It also begins the building of the multicast tree and it stands as the root of this tree. The process of the first phase of multicast tree building is shown in Fig. 1. During the multicast tree building process the transmitting energy needed for data transmission must be minimised. An algorithm to build a multicast tree is shown in Fig. 2.

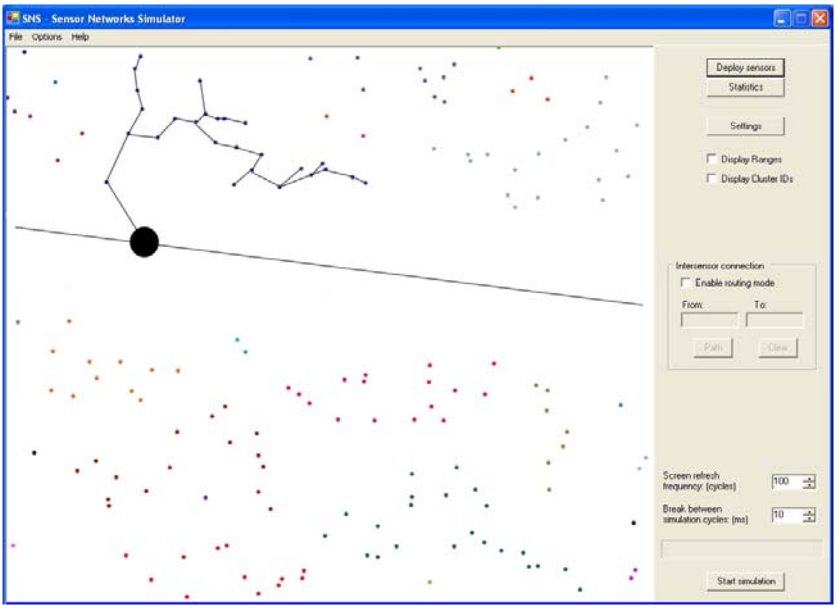



**Fig. 1.** The process of the first phase of multicast tree building

During the second phase, which takes place a moment later, a mobile node sends a packet with a request for data transmission to the built a multicast tree. Since the request to data transmission originates from another position of a mobile node, the other node must be left a multicast tree. Therefore, a multicast tree must be reconfigured. In this phase a mobile node keeps a multicast tree and gathers all the data from the multicast tree. The second phase of multicast routing is realized by a mobile node. Then, during the movement of a mobile node new multicast trees $T_i$, $i = 1, \ldots, n$ arise, for which this node is used as a multicast root.

The implementation of the multicast routing protocol admits two methods. In the first of these, both phases are executed alternately in constant time periods. In the second method of implementation two mobile nodes are used. The first initiates multicast tree building and the second collects all data from the

```
procedure building_ a_ multicast_ tree;
{initialize}
τ_i = {source_node};
for ∀v ∈ V − {τ_i} do
 begin
  P(source_node, v) := admissible_transmit_power;
  set candidate edge to(source_ node, v);
  set candidate edge weight to transmission power to reach v from source node;
 end;
{multicast tree building}
 while (τ_i <> V) do
   begin
    select v ∈ V − {τ_i} with the smallest candidate edge weight;
    add v to τ_i using its candidate edge (u, v);
    increase P(u) to smallest power that reaches v;
    for (v ∈ V − {τ_i}) do
     begin
      select u which minimizes P'(u) − P(u) {P' > P(u) is smallest
                      power to reach v from u}
      set candidate edge to (u, v);
      set candidate edge weight to P'(u) − P(u);
     end;
   end;
```

**Fig. 2.** Pseudo-code of building a multicast tree procedure for mobile sensor network with topology control

multicast tree (see Fig. 3). In our solution the second mobile node must wander close behind the first.

During the *building_ a_ multicast_ tree* procedure a set of nodes belonging to the multicast group is found every time. From this group is selected those which minimize the power of transmission. We assumed that this group is shown by using a gossip algorithm [17]. The activity of this algorithm depends on an exchange of special messages by all members of this group. Let all members of a multicast group be marked from 1 to $n$. We assumed that $m$ nodes ($m < n$) send a message which is a sequential number of the sender. Therefore, each message has an identifier, which defines the global identifier of the sender and the local identifier in the multicast group.

For this communication we defined a table stored in each sensor node. It contains the following sequences, namely:

- a sequence which possesses $m$ elements. The positions in this sequence indicates the number of senders for a given message.
- a sequence with $n$ elements. The positions in this sequence inform us about the actual number of members in the multicast group.
- a sequence containing $n$ elements, which inform us about the time the last message of sender was received.

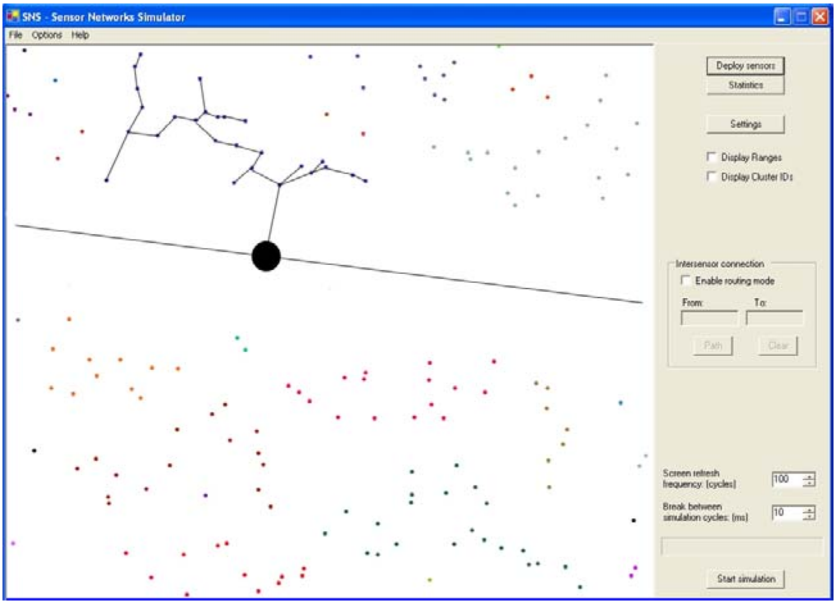



**Fig. 3.** The process of the second phase of multicast tree building

– a sequence possessing $m$ elements. Each position in this sequence defines a number of active sensors in the multicast group. If the multicast group is complete, then all bits in this sequence are equal to 1.
– a sequence which contains $m$ elements. The position of this sequence indicates the number of identifiers of multicast members and contains the number of messages last received.
– a sequence containing $m$ elements. Each position in this sequence is assigned a specific number of groups and contains a number of messages last sent.

The activity of a gossip algorithm begins with the initiating node which sends neighbouring nodes one, two or more messages (see Subsect. 4.1) inviting them to join the multicast group. Each node from the multicast group, which receives an invitation to a multicast group and is not a member of this group, once again sends invitation messages to the remaining members of the group. This procedure continues until $n$-bits sequence of sensor activity contains a defined number of 1. The activity of the gossip algorithm used here is associated with the following messages:

– a *"join-to"* invitation to participate in a multicast group which invites the identifier of the sensor to a created muticast group,
– *"agreement"* – a message which confirms participation in a multicast group,
– *"lack-of-agreement"* – a message which indicates a rejection of an invitation to participate multicast group,
– *"disjoint"* – a message which indicates the liquidation of a multicast group.

# 3 Performance Analysis of MuPMS in Mobile Sensor Networks

In this section, we present an analytical model of multicast group formation using our protocol. Finally, we offer a new measures of multicast protocol for MSNs.

## 3.1 Effectiveness Analysis of Multicast Group Formation in MSNs

The multicast group building can be described by using the epidemic theory which was presented by Bailey [19], Birman [20] and Demers [21]. The formation of a multicast group follows in microsteps. We assumed that each microstep is equal to a time interval in which only one member of a multicast group can send its message.

In effectiveness analysis of multicast group formation in MSNs, we assumed that only one node can initiate this process by sending messages to its neighbours $b$ inviting them to join a multicast group. We can specify two cases: $b = 1$ and $b$ is greater than 1. Both cases will be described in more detail here.

**Model Analysis for $b = 1$.** Let $n$ be the total number of members of a multicast group and $k$ be the number of nodes already invited to a multicast group. We assumed that $P_{\mathrm{a}}$ is the probability that an invitation to a multicast group reaches a specific node before the termination of a given microstep. The process whereby a node is included in a multicast group can be described by the discrete time Markov chain. The state of a system is defined by the number of nodes already invited to multicast group. Let the probability that $k$-th node is invited to a multicast group in microstep $i + 1$ be equal to $P(k_{i+1} = k)$. The probability that in the case of a multicast group containing $n$ members, the number of invited sensor nodes increases by one, is equal to

$$P_{\mathrm{incr}}(k) = \left(\frac{k}{n}\right)\left(\frac{n-k}{n-1}\right)P_{\mathrm{a}} \tag{1}$$

The probability that the number of already invited nodes becomes $k$ in microstep $i + 1$ can be written as

$$P(k_{i+1} = k) - (1 - P_{\mathrm{incr}}(k))P(k_i = k) + P_{\mathrm{incr}}(k-1)P(k_i = k-1) \tag{2}$$

with the following initial conditions: $P(k_i = 0) = 0$, $P(k_0 = 1) = 1$, $P(k_0 = k) = 0$ for $k \neq 1$.

The probability that any members of a multicast group will not be invited to a multicast group after $s$ microsteps can be calculated using the Inclusion-Exclusion Principle [22]. It is equal to

$$P_{\mathrm{lack}}(s) \leq n \cdot P_{\mathrm{lack}}(X, s) = n(1 - P(k_s = n)) \tag{3}$$

where $P_{\text{lack}}(X, s)$ is the probability that any member of a multicast group will not be invited from $V$ members of a multicast group after microstep $s$. The above relationship allows us to computate the number of microsteps needed for to multicast group formation.

**Model Analysis for $b = 3$.** An analysis of the multicast group formation model for $b$ greater than 1 is very complex. The analysis presented below is given for the case $b = 3$. The analysis for other values of $b$ is similar.

It is assumed that only one node can initiate the process of multicast group formation. Each node sends to their neighbours three ($b = 3$) messages inviting them to join the multicast group.

If $k$ out of $n$ members are invited already, then the probability that invitation originated from an already invited node is given by $\frac{k}{n}$. We have here three scenarios in which the number of invited nodes increases by one in a single microstep.

1. One copy of the invited message (*"join-to"*) is sent to an uninvited node, two copies are sent to two already invited members and all copies successfully at their destination. The process can be divided into two stages. In the first, three messages arrive successfully. In the second stage, one message is sent to an uninvited member and two are sent to already invited members. The probability that these events will happen can be written as follows:

$$P_1 = \left(\frac{k}{n}\right)\binom{3}{3}P_{\text{a}}^3\frac{\binom{n-k}{1}\binom{k-1}{2}}{\binom{n-1}{3}} = 3\left(\frac{k}{n}\right)\left(\frac{n-k}{n-1}\right)\left(\frac{k-1}{n-2}\right)\left(\frac{k-2}{n-3}\right)P_{\text{a}}^3 \tag{4}$$

2. One copy of the message *"join-to"* reaches an uninvited member of the multicast group, another copy reaches an invited member and yet another copy is lost. The probability that these events will be happen can be given by

$$\begin{aligned} P_2 &= \left(\frac{k}{n}\right)\binom{3}{3}P_{\text{a}}^2(1-P_{\text{a}})\frac{\binom{n-k}{1}\binom{k-1}{1}}{\binom{n-1}{2}} \\ &= 6\left(\frac{k}{n}\right)\left(\frac{n-k}{n-1}\right)\left(\frac{k-1}{n-2}\right)P_{\text{a}}^2(1-P_{\text{a}}) \end{aligned} \tag{5}$$

3. One copy of the invited message reaches an invited member, the other two copies are lost. The probability that these events will happen is as follows

$$P_3 = \left(\frac{k}{n}\right)\binom{3}{1}P_{\text{a}}(1-Pa)^2\frac{\binom{n-k}{1}}{\binom{n-1}{1}} = 3\left(\frac{k}{n}\right)\left(\frac{n-k}{n-1}\right)P_{\text{a}}(1-P_{\text{a}})^2 \tag{6}$$

The probability that the number of invited members of a multicast group increases by one in one microstep can be written as

$$P_{\text{incr}}(k, 1) = P_1 + P_2 + P_3 \tag{7}$$

We also have here two cases where the number of nodes invited to join a multicast group can increase by two:

1. Two copies of the *"join-to"* message are sent to different members of group. The other copy is sent to an uninvited member. None of these copies are lost. The probability that these events occur is given by

$$\begin{aligned} P_4 &= \left(\frac{k}{n}\right)\binom{3}{3}P_{\mathrm{a}}^3\frac{\binom{n-k}{2}\binom{k-1}{1}}{\binom{n-1}{3}} \\ &= 3\left(\frac{k}{n}\right)\left(\frac{n-k}{n-1}\right)\left(\frac{n-k-1}{n-2}\right)\left(\frac{k-1}{n-3}\right)P_{\mathrm{a}}^3 \end{aligned} \tag{8}$$

2. Two copies of the inviting message *"join-to"* reache different uninvited members of the multicast group successfully and one copy is lost. The probability of these events occuring is given by

$$P_5 = \left(\frac{k}{n}\right)\binom{3}{2}P_{\mathrm{a}}^2(1-P_{\mathrm{a}})\frac{\binom{n-k}{2}}{\binom{n-1}{2}} = 3\left(\frac{k}{n}\right)\left(\frac{n-k}{n-1}\right)\left(\frac{n-k-1}{n-2}\right)P_{\mathrm{a}}^2(1-P_{\mathrm{a}}) \tag{9}$$

Now, we can say the probability that the number of invited members of a multicast group increases by two is equal to

$$P_{\mathrm{incr}}(k,2) = P_4 + P_5 \tag{10}$$

There is only one scenario in which the invitation arrives from three messages. In this scenario all three copies of the message *"join-to"* reach at three different uninvited members successfully. The likelihood that these events will happen is given by

$$\begin{aligned} &P_{\mathrm{incr}}(k,3) = \\ &\left(\frac{k}{n}\right)\binom{3}{3}P_{\mathrm{a}}^3\frac{\binom{n-k}{3}}{\binom{n-1}{3}}\left(\frac{k}{n}\right)\left(\frac{n-k}{n-1}\right)\left(\frac{n-k-1}{n-2}\right)\left(\frac{n-k-2}{n-3}\right)P_{\mathrm{a}}^3 \end{aligned} \tag{11}$$

The probability that the number of invited members in microstep $i+1$ is $k$ $(0 < k \leq n)$ consists of four parts. The first part concerns a case in which the number of invited nodes in $i$-th microstep is equal to $k$ and does not increase. The second part comes from the case where the number of invited members of a multicast group is equal to $k-1$ and increases by one. The third part concerns a case in which the number of invited members is equal to $k-2$ and increases in one microstep by two, etc. Thus, the probability of these events occuring is equal to

$$\begin{aligned} P(k_{i+1} = k) = {} & (1 - P_{\mathrm{incr}}(k,1) - P_{\mathrm{incr}}(k,3) - P_{\mathrm{incr}}(k,3))(P(k_i = k) \\ & + P_{\mathrm{incr}}(k-1,1)P(k_i = k-1) + P_{\mathrm{incr}}(k-2,2)P(k_i = k-2) \\ & + P_{\mathrm{incr}}(k-3,3)P(k_i = k-3) \end{aligned} \tag{12}$$

There are here two special cases when $k = 1$ and $k = 2$. Thus, initially one node is invited. The probability that one node is invited to a multicast group in microstep $i+1$ is available from the case in which no node has been invited in this microstep. The probability of these events occuring is equal to

$$P(k_{i+1} = 1) = (1 - P_{\text{incr}}(1,1) - P_{\text{incr}}(1,2) - P_{\text{incr}}(1,3)) \cdot P(k_i = 1) \qquad (13)$$

Analogously, the probability that two members of a multicast group will be invited in microstep $i+1$ comes from either of two cases in which no node gets invited or just one node gets invited. The probability for these events is equal to

$$P(k_{i+1} = 2) = (1 - P_{\text{incr}}(1,1) - P_{\text{incr}}(1,2) - P_{\text{incr}}(1,3)) \cdot P(k_i = 1) \qquad (14)$$

Initially only one node is invited to the multicast group. Thus, the initial conditions for this Markov chain are as follows: $P(k_i = 0) = 0, \ldots$. The probability that all the $n$ nodes are invited after $s$ microsteps, $P(k_s = n)$, can be computed in the following way. As with the subset size 1 case, the probability that no node gets invited by the sequence number information from node $X$ after $s$ microsteps is equal to $P_{\text{incomp}}(X,s) = 1 - P(k_s = n)$. This probability is bounded by

$$P_{\text{incomp}}(s) \leq n \cdot P_{\text{incomp}}(X,s) = n(1 - P(k_s = n)) \qquad (15)$$

The complexity of the presented method is equal to $O(\log n)$ where $n$ is the number of the multicast group size. This means that the number of microsteps is of the order of $O(n \log n)$.

### 3.2 Energy-Efficiency Analysis

In this subsection, we study energy-efficiency of our model as compared with the flat ad hoc network.

Assume that $n$ sensors are randomly and uniformly deployed on a disk of radius $R$ which covered all sensors in the multicast group. The node density in the multicast group is given by $n/\pi R^2$. We analyzed here the energy cost of sending one packet originated at a randomly chosen sensor to a root. We use the radio model considered in [23]. When a sensor is receiving a packet, it consumes $E_{\text{rx}}$ [Joule/bit]. A transmission that covers a neighborhood of radius $r$ consumes $E_{\text{tx}}(r)$ [Joule/bit], which is given by

$$E_{\text{tx}}(r) = e_{\text{tx}} + \max\{e_{\min}, e_{\text{out}} r^{\alpha}\} \qquad (16)$$

where $e_{\text{tx}}$ is the energy consumed by the transmitter circuitry, $e_{\min}$ is the minimum energy radiated regardless of the transmission range, $e_{\text{out}}$ is the antenna output energy to reach and $\alpha$ is the path attenuation factor. We note that $e_{\min}$ gives a hard limit on the minimum transmission range, namely

$$r \geq r_0 \stackrel{\triangle}{=} \left(\frac{e_{\min}}{e_{\text{out}}}\right)^{\frac{1}{\alpha}} \qquad (17)$$

Let $X$ be the distance from chosen sensor to the root node. The probability density function (pdf) of $X$ is given by

$$p_X(x) = \frac{2}{R^2}x, \quad 0 < x \le R \tag{18}$$

Thus, the total energy $E_{\text{AdHoc}}(r)$ consumed by sending one packet to the root node with an optimal transmission range $r$ is given by

$$E_{\text{AdHoc}} = \min_{r \ge r_{\min}} \frac{2}{R^2} \int_0^R E(r)h(x,r)xdx \tag{19}$$

where $r_{\min}$ is the minimum transmission range to ensure node connectivity, $E(r)$ is the energy consumed in one hop, $h(x,r)$ is the number of hops for a packet to reach the root node $x$ meters away.

We recall that a sufficient condition for network connectivity for large $n$ is equal to $\frac{r^2}{R^2} = O(\log n/n)$ [24]. Thus, we have

$$r_{\min} = \max\left\{r_0, R\sqrt{\frac{\log n}{n}}\right\} \tag{20}$$

We assume that a sensor in multicast group has, on the average, $(n-1)(r^2/R^2)$ neighbours who listen to that sensor's transmission. Thus, the energy consumed in one hop is given by

$$E(r) = E_{\text{tx}}(r) + (n-1)\frac{r^2}{R^2}E_{\text{rx}} \tag{21}$$

Thus, from Eqs. (19) – (21) the total energy consumed by sensors in the flat ad hoc multicast group is given by [25]

$$E_{\text{AdHoc}} = \min_{r \ge r_{\min}} \frac{2R}{3r}\left(E_{\text{tx}}(r) + (n-1)\frac{r^2}{R^2}E_{\text{rx}}\right) \tag{22}$$

The optimal transmission range in the asymptotic regime $n \to \infty$ is given by $r_{\min}$ (see Eq. (20)).

In the protocol MuPMS, the mobile root positions itself at a random location and broadcasts a packet to activate sensors in its converage area. Let $\theta_0$ denote the minimum angle of the coverage area for the multicast group of sensors. The total energy consumed by sensors in the multicast group is given by

$$E_{\text{MuPMS}} = \min_{\theta \ge \theta_0} \frac{d^2 \tan^2\theta}{R^2} n \cdot E_{\text{rx}} + E_{\text{tx}}(d) \tag{23}$$

where $d$ is the transmission range in meters from the last sensor then transmits directly to the mobile root. Comparing Eq. (22) with Eq. (23), we see that when the multicast group size $n$ is increaed by $R$, MuPMS offers orders of magnitute of improvement over the flat ad hoc architecture in energy efficiency.

### 3.3 Performance Measures of MuPMS

To analyse the performance of MuPMS we introduced the following measures:

1. *Ratio of Data Delivery.* This is the quotient of a number of delivered data packets to all packets received by receivers. It is the ratio of the effectiveness of the protocol.
2. *Ratio of Data Forwarding Efficiency.* This is the number of data packet transmissions per delivered packet. This measure takes into consideration the number of discarded and retransmitted packets. In the case of our protocol this measure is less than 1.
3. *Ratio of the Number of Sent Bytes in Controlling Packets to the Number of Data Bytes Received by Receivers.* This measure defines the overheads connected with network control.
4. *Ratio of the Number of Data Bytes for Control and Receivers to the Amount of Data Received by the Nodes.* This measure defines the data channel effectiveness, which is important in wireless communication.
5. *Ratio of the Energy Expenditure of the Flat Ad Hoc to that of the MuPMS,* $E_{\mathrm{AdHoc}}/E_{\mathrm{MuPMS}}$*, as a Function of the Node Density* $\rho$. This measure defines the gain in energy efficiency achieved by MuPMS protocol.

## 4 Experimental Results

In this section, we present a simulation and the numerical results of a performance evaluation of the MuPMS protocol.

Our simulation network is 500 m × 500 m in size. This sensor field includes both mobile and immobile sensor nodes. We assumed that the ratio of immobile to mobile sensor nodes, $\xi$, can change in simulation experiments from 0.5 to 1. The radio transmission range of each node varies from 10 m to 35 m. The movement of each mobile node follows a random waypoint model. Moreover, the speed of mobile nodes can change from $v_{\min}$ to $v_{\max}$.

In the first simulation, we chose the multicast group size to be 5 up to 60. For all multicast sessions we have observed the following metrics: (a) *relative tree cost*, and (b) *relative maximum delay*. There, the relative tree cost is the sum of the physical hop lengths of all wireless links of the multicast tree. We assumed that the relative maximum delay is the number of physical hops along the longest path from the source to any of the receivers on the tree. By way of comparison we computed the optimal tree cost from a Steiner tree built on the physical topology for the same number of nodes in the multicast group. The optimal maximum delay was calculated on the Shortest Path Tree built on a physical topology with the same root.

Figure 4 shows the relative tree cost versus the multicast group size for the ratio $\xi$ which is equal to 0.5 and 0.8, respectively. As is shown in the graphs in Fig. 4(a), MuPMS builds more efficient multicast trees at lower mobile sensor node speeds than at higher speeds. This is because at network topology at

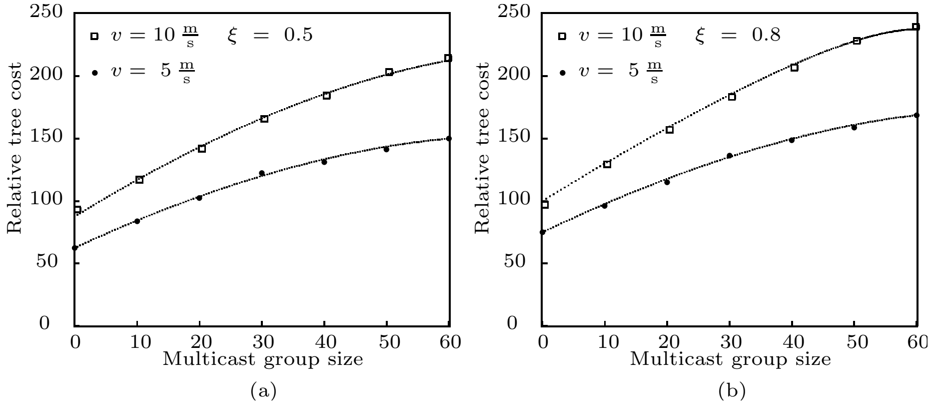



**Fig. 4.** Relative tree cost as function of the multicast group size in mobile sensor network for ratio $\xi = 0.5$ and 0.8, respectively

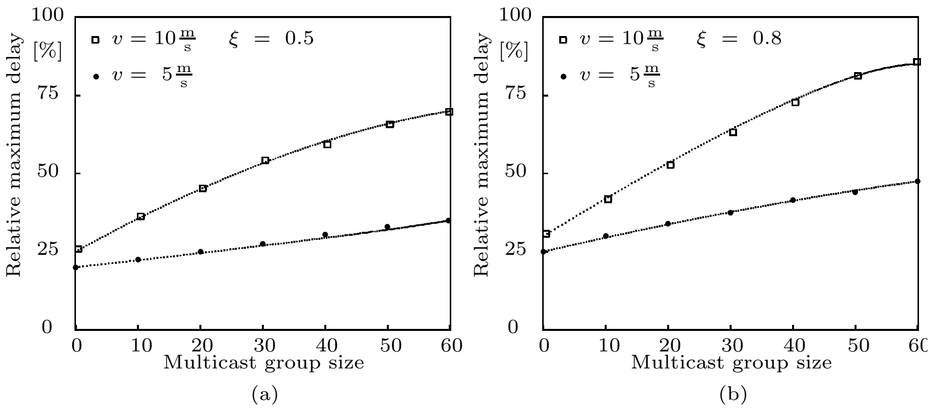



**Fig. 5.** Relative maximum delay as function of the multicast group size in mobile sensor network for ratio $\xi = 0.5$ and 0.8, respectiely

a higher speed is more complicated and the relative tree cost is greater than for lower sensor node speeds.

Figure 5 shows the relative maximum delay versus the multicast group size for different values of ratio $\xi$ which are 0.5 and 0.8, respectively. As is shown in the graph in Fig. 5(a) the relative maximum delay increases according to the multicast group size and also increases according to the speed of the mobile sensor nodes. The same notice is observed for higher values of the $\xi$ parameter.

The main effectiveness parameters of the MuPMS protocol are plotted in Fig. 6. As shown in the graph in Fig. 6(a), the Ratio of Data Delivery increases with the speed of the mobile sensor nodes. This is because the nodes can route to other nodes earlier, when they all are moving. Data Forwarding Efficiency (see Fig. 6(b)) shows that the volume of data packet transmission per delivered packet

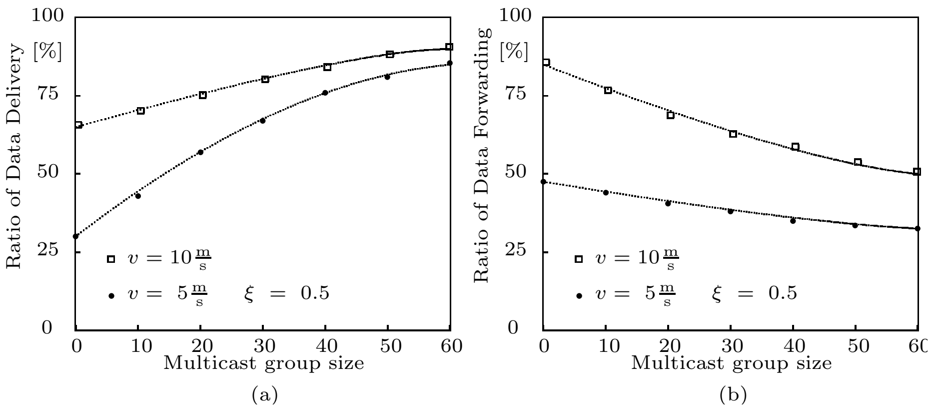



**Fig. 6.** Ratio of Data Delivery and Ratio of Data Forwarding as function of the multicast group size in mobile sensor network for ratio $\xi = 0.5$

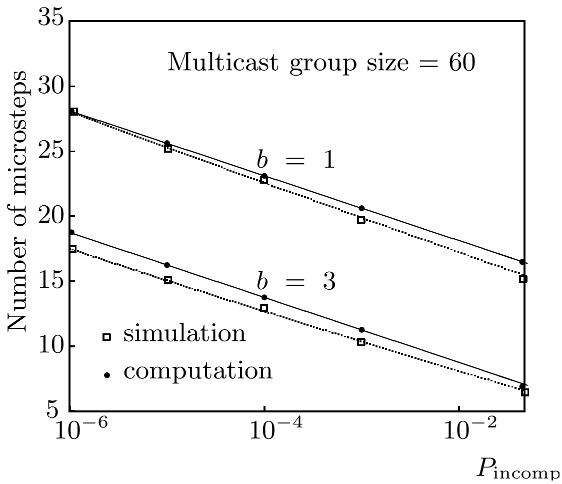



**Fig. 7.** Number of microsteps as function of the probability of incompleteness $P_{\text{incomp}}$ of multicast group

indicates that the higher speed causes an increase of transmission overheads and thus decreases the value of this parameter.

The multicast group building process was observed during the execution of the *formation_ of_ multicast_ group* procedure. This procedure causes a single *"join-to"* message to be sent. Therefore, the number of microsteps needed to form a multicast group is equal to the number of microsteps divided by the number of nodes in a multicast group. Figure 7 shows the results of computing and simulating the number of microsteps needed to formation a multicast group in an MSN, consisting of 60 nodes depending on the value of $P_{\text{incomp}}$ – the probability that no node in this group was invited to join a multicast group. It

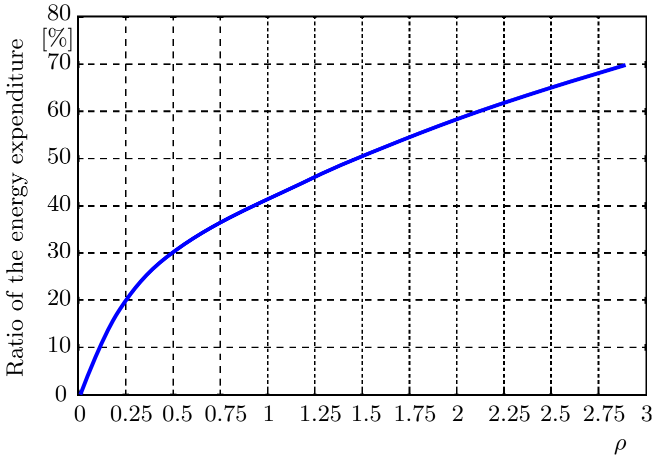



**Fig. 8.** Ratio of the energy expenditure of the flat ad hoc to that of the MuPMS as a function of the node density $\rho$ ($R = 200\,\mathrm{m}$, $\alpha = 3$, $d = 50\,\mathrm{m}$, $r_0 = 10\,\mathrm{m}$, $d \tan\theta_0 = 10\,\mathrm{m}$, $e_{\mathrm{tx}} = 80\,\mathrm{nJ}$, $e_{\mathrm{rx}} = 180\,\mathrm{nJ}$)

is evident that the increase in copies of invitation from $b = 1$ to $b = 3$ decreases the number of microsteps needed to about ca. 30%.

The scalability of our protocol is univocal. As is shown in the simulations, the duration of time between the start of the stability detection protocol and the joint moment of the first node (Time Per Round, TPR) increases together with varying group sizes. On the other hand, the number of microsteps used to form a multicast group increases linearly together as the group size change to a defined value. When the multicast group is large it does not increase.

We next consider the comparison between the flat ad hoc network and MuPMS protocol. Figure 8 depicts a numerical result on the ratio of the energy expenditure of flat ad hoc to that of MuPMS as a function of the node density $\rho$. We see that MuPMS offers improvement which increases with the size of multicast group. In our analysis the energy consumed by the mobile root is not included.

## 5 Conclusion

In this paper, we introduced a new multicast routing (MuPMS) protocol for mobile sensor networks. The protocol allows us to harvest data using mobile nodes which move in the sensor field. The MuPMS protocol possesses comparable parameters as known protocols to mobile ad hoc networks. Moreover, it tolerates messages loss without requiring a reliable multicast protocol underneath. This scheme overcomes routing errors and link failures, because messages are randomly sent to other sensor nodes of the multicast group. It can also tolerate group membership changes caused by member crashes or departures. This is possible by using a similar technique applied in routing table updates.

In future work, we will develop a more reliable version of the MuPMS protocol. For implementation in mobile networks with higher node speeds. It can be applied to a multicast in mobile networks based on cars or airplanes.

## References


1. Akyildiz, I.F., Su, W., Sakarasubramaniam, Y., Cayirci, R.: Wireless Sensor Networks: A Survey. Computer Networks 38, 393–422 (2002)
2. Tan, J., Ning-Xi: Integration of Sensing, Computing, Communication and Cooperation for Distributed Mobile Sensor Networks. In: Proceedings of the IEEE Int. Conference on Robotics, Intelligent Systems and Signal, vol. 1, pp. 54–59 (2003)
3. Joengmin-Hwang, Du, D.H.C., Kusmierek, E.: Energy Efficient Organization of Mobile Sensor Networks. In: Proc. IEEE Int. Conf. Parallel Proces., vol. 2, pp. 84–91 (2004)
4. Shu-Zhou, Min-You-Wu, Wei-Shu: Terrain-Constrained Mobile Sensor Networks. In: Proc. of the IEEE GLOBECOM, p. 5 (2005)
5. Nojeong-Heo, Varshney, P.K.: Energy-Efficient Deployment of Intelligent Mobile Sensor Networks. IEEE Trans. Systems, Man, Cyb., Part A 35(1), 78–92 (2005)
6. Gangadharpolli, S., Golwelkar, U., Varadarjan, S.: A Topology Based Localization in Ad Hoc Mobile Sensor Networks. In: Battiti, R., Conti, M., Cigno, R.L. (eds.) WONS 2004. LNCS, vol. 2928, pp. 16–28. Springer, Heidelberg (2004)
7. Sammer-Tilak, Vinay-Kolar, Abu-Ghazaleh, N.B., Kang, K.D.: Dynamic Localization Control for Mobile Sensor Networks. In: Proceedings of the IEEE Int. Performance Computing and Communications Conference, pp. 587–592 (2005)
8. Dharme, A.G., Jaeyong-Lee, Jayasuriya, S.: Using Fuzzy Logic for Localization in Mobile Sensor Networks: Simulations and Experiments. In: American Control Conference, 6 (2006)
9. Curino, C., Giani, M., Giogtta, M., Gusti, A., Murphy, A.L., Picco, G.P.: TinyLIME: Bringing Mobile Sensor Networks Through Middleware. In: Proceedings of the Third IEEE Int. Conf. on Pervasive Computing and Communications (PERCOM), pp. 60–72 (2005)
10. Friedman, J., Lee, D., Tsigkogiannis, I., Wong, S., Chao, D., Levin, D., Kaiser, W., Srivastava, M.: Ragobot: A New Platform for Wireless Mobile Sensor Networks. In: Prasanna, V.K., Iyengar, S.S., Spirakis, P.G., Welsh, M. (eds.) DCOSS 2005. LNCS, vol. 3560, pp. 412–412. Springer, Heidelberg (2005)
11. Paul, S., Sabnani, K.K., Lin, J.C., Bhattacharyya, S.: Reliable Multicast Transport Protocol (RMTP). IEEE Journal on Selected Areas in Communications 15(3), 407–421 (1997)
12. Chandra, R., Ramasubramanian, V., Birman, K.: Anonymous Gossip: Improving Multicast Reliability in Mobile Ad Hoc Networks. In: Int. Conf. on Distributed Computing Systems, pp. 275–283 (April 2001)
13. Luo, J., Engster, P.T., Hubaux, J.-P.: Route Driven Gossip: Probabilistic Reliable Multicast in Ad Hoc Networks. In: Proc. of the IEEE INFOCOM, San Francisco, CA, pp. 2229–2239 (2003)
14. Tang, K., Obraczka, K., Lee, S.J., Gerla, M.: A Reliable, Congestion-Controlled Multicast Transport Protocol in Multimedia Multi-Hop Networks. In: Proceedings of the IEEE WPMC, Honolulu, pp. 252–256 (2002)
15. Obraczka, K.: Multicast Transport Mechanism: A Survey and Taxonomy. IEEE Trans. on Communication Magazine 36(1), 94–102 (1998)

16. Haccoun, D., Pierre, S.: Automatic Repeat Request. In: Gibson, J.D. (ed.) The Communications Handbook, pp. 181–198. CRC Press/IEEE Press, Boca Raton (1996)
17. Haas, Z.J., Halpern, J.Y., Li, L.: Gossip-Based Ad Hoc Routing. Proc. of the IEEE INFOCOM 3, 1707–1716 (2002)
18. Vijay, K., Bhargava, I., Fair, J.: Forward Error Correction Coding. In: Gibson, J.D. (ed.) The Communications Handbook, pp. 166–180. CRC Press/IEEE Press, Boca Raton (1996)
19. Bailey, N.T.J.: The Mathematical Theory of Infections Diseases and Its Applications, 2nd edn. Hafner Press (1975)
20. Birman, K.P., Hayden, M., Ozkasap, O., Xian, Z., Budiu, M., Minski, Y.: Bimodal Multicast. ACM Transaction on Computer Systems (May 1999)
21. Demers, A., Greene, D., Hausner, C., Irish, W., Larson, J., Shenker, S., Sturgis, H., Swinehart, D., Terry, D.: Epidemic Algorithms for Replicated Database Maintenance. In: Proceedings of the ACM Symposium on Principles of Distributed Computing, Vancouver, pp. 1–12 (1987)
22. Cormen, T.H., Leisersen, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, 2nd edn. MIT Press, Cambridge (2001)
23. Shih, E., Cho, S., Ickes, N., Min, R., Sinha, A., Wang, A., Chandrakasan, A.: Physical Layer Driven Protocol and Algorithms Design for Energy-efficient Wireless Sensor networks. In: Proc. ACM MOBICOM, Rome, Italy, pp. 272–286 (July 2001)
24. Gupta, P., Kumar, P.R.: The Capacity of Wireless Networks. IEEE Trans. on Inf. Theory 46(2), 388–404 (2000)
25. Zhao, Q., Tong, L.: Energy Efficiency of Large-scale Wireless Networks: Proactive vs. Reactive Netorking. IEEE J. on Sel. Areas Commun. 23(5), 1100–1112 (2005)

# On the Lifetime Maximization Design of Mobile Ad Hoc and Sensor Networks

Jerzy Martyna

Institute of Computer Science, Jagiellonian University, ul. Łojasiewicza 6, 30-348 Cracow, Poland
martyna@softlab.ii.uj.edu.pl

**Abstract.** One critical issue in the design of mobile ad hoc and sensor networks is to select some parameters that can guarantee a longer time period during which these networks are fully working. To maximize this period, which is called the *network life-time*, a new method is proposed. For this purpose, a radio link availability estimate as well as a routing parameters, such as the movement directions, speed, actual interference, path availability, etc., are used. With this method an optimal assignment of mobile nodes to clusterheads at each time step was obtained. The given approach allows to maximize only the minimum lifetime without creating topology or also to create the topology after having done the previous work. We validated our approach through simulation and showed that the proposed framework is well suited for the designing of mobile ad hoc and sensor networks.

## 1 Introduction

Wireless mobile ad hoc networks [1] are the ultimate frontier in wireless communication. This technology allows us to build a network without the need for a fixed infrastructure. Moreover, by exploiting ad hoc networks, various portable devices (cellular phones, PDA, laptops, and so on) and fixed equipment (access points, base stations, etc.) can be connected together.

Wireless sensor networks are a particular type of ad hoc networks [14] in which the nodes act as 'smart sensors'. In these networks, all nodes are equipped with advanced sensing functionalities, a small processor and a short-range wireless transceiver. Each node has the ability to sense elements of its environment, perform simple computations, and communicate either among its peers or directly to an external sink.

An important issue in ad hoc and wireless sensor networks is the design of these networks. The limited energy and bandwidth resources [7] require substantial effort from the designers. Currently, the design of these networks depends on the skill of the designers' decisions regarding their movements, network topology and transmitting ranges. Moreover, the problem of maximizing network lifetime, which is defined as the time until it takes for the first node to run out of battery power [5], becomes one of the most critical performance measures.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 50–61, 2009.

Other definitions of ad hoc and wireless sensor network lifetime have been suggested in the literature. These definitions include the time it takes until a fraction of sensors run out of battery [7], the average network lifetime [17], the time until the first loss of some of the desired coverage [2]. The lifetime of a wireless sensor network as well as an ad hoc network is also defined as the time after which the first node (link) disconnects [5], [10]. The definition is especially useful for determining the lifetime of these networks functioning in real-time applications. It is also referred to as the *worst-case lifetime* model. The lifetime limits of energy-constrained wireless sensor networks was also determined by Hu [8]. A survey of the problem of maximizing network lifetime at various levels of energy consumption models is successfully achieved in a paper by Dong [6].

Node mobility is a prominent feature of ad hoc networks and also in wireless sensor networks. The most important mobility model used in the simulation and design solutions is the Random Waypoint (RWP) model [9], [4], in which each node chooses uniformly at random a destination point (the 'waypoint') within the deployment region $R$, and moves toward it along a straight line. Node speed is chosen uniformly at random in the given interval. When the node arrives at destination, it remains stationary for a predicted pause time, and then starts moving again according to the same pattern. In this paper, we assume an environment where each mobile node has the RWP model.

Several researchers have proposed design solutions of mobile ad hoc and wireless sensor networks. Among others, an established routing path in the mobile ad hoc networks was suggested by S.Y. Wang [18]. This approach compares the performances of a local and a global path-repair design in these networks. Boukerche [3] presented two algorithm design techniques. Firstly, the algorithm creates and maintains routing paths among the nodes. In the second algorithm a randomly moving subset of the nodes acting as an intermediate pool for receiving and delivering messages was given. Another method for cross-layer designs in mobile ad hoc networks was presented by Xinsheng-Xie [19]. In this approach a fuzzy logic system for coordinating physical-, data-link layer and application layer designs was used. Among others, by using this method the average delay was minimized and the network lifetime was increased. A power conservative cross layer design for mobile ad hoc networks with some performance metrics was presented by Ramachandran [13]. Ki-II-Kim [12] suggested a new multicast protocol in mobile ad hoc network design.

Our goal is to find a novel method for designing mobile ad hoc and wireless sensor networks. This method is based on the network (clusterheads) lifetime of the designed mobile ad hoc and sensor networks. In the lifetime of a network computation we have used a radio link availability estimate as well as a routing metric, such as the movement directions, speed, actual interference, path availability, etc. With the help of the method proposed here we obtained the optimal assignment of mobile nodes to clusterheads at each time step.

The rest of the paper is organized as follows. First, we give a detailed description of the estimation of the radio link availability. In Sect. 3, we present the proposed solution. Then, we outline our algorithm of the mobile network design. We present the experimental results in Sect. 4. Finally, Sect. 5 provides the concluding remarks.

## 2 The Radio Link Availability Estimation in the Mobile Ad Hoc and Sensor Networks

We assume that each node of a mobile ad hoc and sensor network has the possibility to estimate a random length interval during which a node moves in a constant direction at a constant speed. We define this time length interval as the mobility epoch length. We further assume that mobility epoch lengths are exponentially distributed with mean $\lambda^{-1}$, namely

$$\begin{aligned} E(x) &\stackrel{\triangle}{=} P\{\text{time length } \leq x\} \\ &= 1 - e^{-\lambda x} \end{aligned} \tag{1}$$

Assuming that the node mobility is uncorrelated, we can define the availability [11] as

$$L(T_p) \stackrel{\triangle}{=} P\{t_0 + T_p \mid \text{available at } t_0\} \tag{2}$$

which gives the probability that the link will be continuously available from time $t_0$ to $t_0 + T_p$.
The calculation of $L(T_p)$ can be achieved in two parts, namely

$$L(T_p) = L_1(T_p) + L_2(T_p) \tag{3}$$

The first term indicates the speed of the two nodes which is unchanged between $t_0$ and $t_0 + T_p$. It is equal to the probability that the time length intervals from $t_0$ to $t_0 + T_p$ are longer than $T_p$ because $T_p$ is an accurate prediction if the movement of two nodes is unchanged [16].

Since the nodes' movements are independent of each other and with the exponential distribution, then $L_1(T_p)$ is defined as

$$\begin{aligned} L_1(T_p) &= (1 - E[T_p])^2 \\ &= e^{-2\lambda T_p} \end{aligned} \tag{4}$$

The radio link availability of link $L_2(T_p)$ is devoted to all other cases. The calculation of this probability is more complicated because of the difficulties in the changes in the radio link status caused by changes in a node's movement.

Let be $\phi < T_p$ is a random variable for the time interval between $t_0$ and $t_0 + T_p$ during which either of two nodes or both change their movements. Thus, $P\{\phi \leq \Phi < T_p\}$ denotes the probability that the movement of both nodes is unchanged between the time interval $t_0$ and $t_0 + \phi$ [15]. This probability is given by

$$\begin{aligned} P\{\phi \leq \Phi < T_p\} &= 2\,[E(T_p) - E(\phi)]\,[1 - E(T_p] \\ &\quad + [E(T_p - E(\phi)]^2 \\ &= e^{-2\lambda\phi} - e^{-2\lambda T_p} \end{aligned} \tag{5}$$

To estimate the radio link availability corresponding to $\phi$ we compute $\mathcal{L}(\phi)$ as follows

$$\mathcal{L}(\phi) = \frac{\phi + (T_p - \phi)pe^{-2\lambda(T_p-\phi)}}{T_p} + \epsilon \tag{6}$$

where $p$ is the probability that the distance of two moving nodes after changing their movements is closer, $\epsilon$ ($\epsilon > 0$) is an adjustment to the radio link availability. Both values are independent of $\phi$.

To estimate $L_2(T_p)$ we use the average $\mathcal{L}_{\in}(\phi)$ over $\phi$, namely

$$\overline{\mathcal{L}}_2 = \int_0^{T_p} \mathcal{L}_2(\phi) f(\phi) d\phi \tag{7}$$

where $f(\phi) \geq 0$ is given by

$$\begin{aligned} f(\phi) &= \lim_{\delta\phi \to 0} \frac{P(\phi \leq \Phi < T_p) - P(\phi \leq \Delta\phi < \Phi < T_p)}{\delta\phi} \\ &= -\frac{dP\{\phi \leq \Phi < T_p\}}{d\phi} \\ &= 2\lambda e^{-2\lambda\phi} \end{aligned} \tag{8}$$

Substitute $\mathcal{L}_2(\phi)$ and $f(\phi)$ in Eq. 7 with Eqs. 6 and 8, respectively. Thus, the value of $\overline{\mathcal{L}}_2$ can be calculated by

$$\begin{aligned} \overline{\mathcal{L}}_2 &\approx \int_0^{T_p} \left( \frac{\phi + (T_p - \phi)pe^{-2\lambda(T_p-\phi)}}{T_p} + \epsilon \right) \cdot 2\lambda e^{-2\lambda\phi} d\phi \\ &= \frac{2\lambda}{T_p} \int_0^{T_p} [\phi e^{-2\lambda\phi} + pe^{-2\lambda T_p}(T_p - \phi) + \epsilon T_p e^{-2\lambda\phi}] d\phi \\ &= \frac{1}{2\lambda T_p} + \epsilon + e^{-2\lambda T_p} \left( p\lambda T_p - \frac{1}{2\lambda \cdot T_p} - \epsilon - 1 \right) \end{aligned} \tag{9}$$

An estimation of $L(T_p)$ can be obtained by

$$\begin{aligned} L(T_p) &\approx L_1(T_p) + \mathcal{L}_2 \\ &= \frac{1}{2\lambda T_p} + \epsilon \\ &\quad + e^{-2\lambda T_p} \left( p\lambda T_p - \frac{1}{2\lambda T_p} - \epsilon \right) \end{aligned} \tag{10}$$

The value of $\epsilon$ depends on the environment factors, such as spatial node density, node's radio coverage, etc. From the measures [17] the value of $\epsilon$ is between 0.28 and 0.53. In the case of uncertainty $\epsilon$, it is fine to set $\epsilon = 0$.

## 3 The Design of Mobile Wireless Ad Hoc and Sensor Networks

In this section, we describe our method of designing the mobile ad hoc and sensor networks.

We assume that the system parameters are known: the number of the moving nodes in the network ($N_{\mathrm{M}}$), the number of clusterheads in the network ($N_{\mathrm{C}}$), the value of battery energy at each clusterhead ($E^{\mathrm{battery}}$).

We assume that only the clusterhead nodes possess the possibility to connect with the mobile nodes. All static ordinary nodes can transmit data to their clusterhead nodes.

Let $d_{ik}$ be the Euclidean distance between clusterhead $i$ and mobile node $k$ $(i = 1, 2, \ldots, N_{\mathrm{C}}; k = 1, 2, \ldots, N_{\mathrm{M}})$. Let the radius of cluster $i$ be equal to $r_i = d_{ij}$ when $j$ is the farthest mobile node controlled by clusterhead $i$. Given a prediction $T_p$ on the continuously available time for a radio link between the clusterhead $i$ and the mobile node $k$ in time $t_0$, the radio link availability of the link $d_{ij}$ is given by

$$L_{ij}(T_p) = L(T_p) \cdot d_{ij} \tag{11}$$

Let $\tau_{ij}$ describe the lifetime of clusterhead $i$ with the radius $r_i^s = d_{ij}$ at time step $s$. Each cluster contains $n_{ij} = \{k \in N_{\mathrm{M}} \mid d_{ik} < d_{ij}\}$ mobile nodes.

We introduce matrix $\tau = \{\tau_{ij}\}$ whose dimension is equal to $\mid N_{\mathrm{C}} \mid \times \mid N_{\mathrm{M}} \mid$ and where each element is given by

$$\tau_{ij} = \frac{E_i^{\mathrm{battery}}}{\alpha(L_{ij}(T_p))^2 + \beta \mid n_{ij} \mid} \tag{12}$$

where $\alpha$ and $\beta$ are constant weight factors.

Assuming that the network lifetime is limited by the clusterheads' functioning time, the network lifetime can be defined as [5]

$$\tau_{\mathrm{net}} = \min_{i \in N_{\mathrm{C}}} \{\tau_{ij}\}, \quad j \in N_{\mathrm{M}} \tag{13}$$

Let the optimal assignment of mobile nodes to clusterheads at time step $s$ be described by the binary variable $x_{ij}^s$. We assumed that the value of $x_{ij}^s$ is equal to 1 if clusterhead $i$ covers mobile node $j$ at time step $s$ and equal to 0 otherwise.

Thus, we can formulate for each $i$ $(i = 1, 2, \ldots, N_{\mathrm{C}})$ at each time step $s$ $(s \geq T_p)$ the following maximization problem

$$\begin{array}{lll} \text{maximize} & \tau_{\mathrm{net}}^s & \\ \text{subject} & \sum_{i=1}^{N_{\mathrm{C}}} x_{ij}^s \geq 1 & \forall j \in N_{\mathrm{M}} \\ & \tau^s \leq \tau_{ij}^s \cdot x_{ij}^s + m(1 - x_{ij}^s) & \forall i \in N_{\mathrm{C}}, \quad j \in N_{\mathrm{M}} \\ & x_{ij}^s \in \{0, 1\}, \quad \tau^s \geq 0, & \forall i \in N_{\mathrm{C}}, \quad j \in N_{\mathrm{M}} \end{array} \tag{14}$$

where $m$ is a sufficiently large value.

```
procedure create_topology;
begin
 max_lifetime := 0.0;
 for j := 1 to N_M do
  for i := 1 to N_C do
   if τ_ij ≥ max_lifetime then
    max_lifetime := τ_ij;
    find(i) for given max_lifetime;
   endif;
   cover sensors j with clusterhead i;
  endfor;
 endfor;
end;
```

```
procedure mobile_network_design;
begin
 for i := 1 to N_C do
  check E_i^battery;
  for j := to N_M do
   compute d_ij, L_ij(T_p), | n_ij |;
   compute τ_ij;
  endfor;
 endfor;
 find τ^s_net = min_{i∈N_C}{τ_ij};
 for s := 1 to T do
  create_topology;
  maximize τ^s_net;
 endfor;
end;
```

**Fig. 1.** The pseudo-code of the mobile network design algorithm

The first constraint concerns the requirement that each mobile node is covered by one clusterhead at least. The second constraint states that if mobile node $j$ is assigned to clusterhead $i$, the system cannot live more than $\tau_{ij}^{s}$. If mobile node $j$ is not assigned to clusterhead $i$, the last constraint is relaxed by taking a sufficient value of $m$.

It can be stated that the model parameters depend on the time step. Nevertheless, the summarized value of $\tau_{\text{net}}^{s}$ allows us to find the total lifetime of the whole network, namely

$$\tau_{\text{net}} = \sum_{s=1}^{T} \tau_{\text{net}}^{s} \tag{15}$$

where $T$ is an admissible time horizon value.

The pseudo-code of the mobile ad hoc and sensor network design algorithm is given in Fig. 1.

Finally, we note the other result of the algorithm given in Fig. 1. If the only objective is to maximize only the minimum lifetime without creating a topology, then we need only to apply the first two loops in a *mobile_network_design* procedure. Hence, according to our formulation, only one minimization of the lifetime is needed to maximize the minimum lifetime.

## 4 Simulation Results

In this section we evaluate the effectiveness of our method for designing mobile wireless ad hoc and sensor networks. The simulation is run using the own simulation program for the designing of a mobile ad hoc and sensor networks. The toolkit of this program regarding the radius of clusterhead $r_i = 50\,m$ is given in Fig. 2.

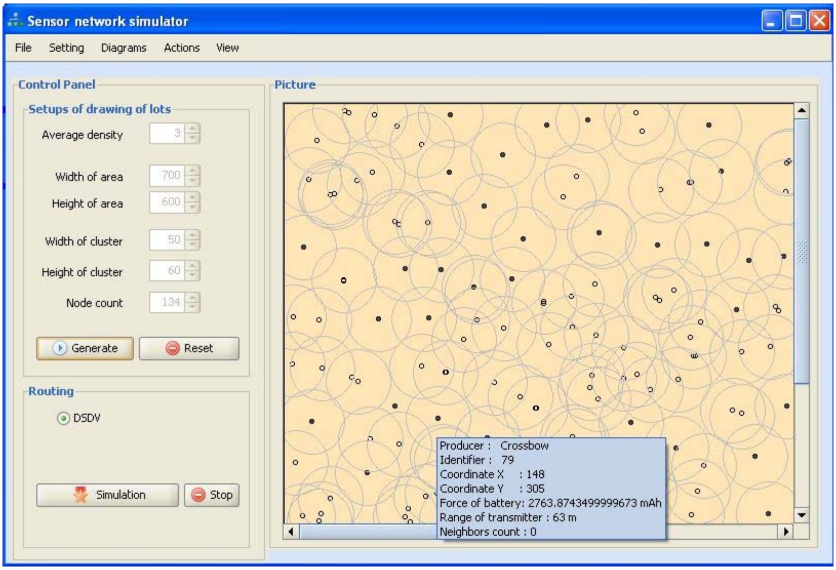



**Fig. 2.** A toolkit of our simulation program

The simulation results are obtained for a two-dimensional space with mobile and immobile nodes. The maximal radius of a mobile's radio coverage is up to 100 meters. The transmission rate of a mobile node is 2 Mbps. Only the exponentially distributed time intervals have been investigated. The simulation time is equal to 1000 s. The maximal radius of a mobile node is 2 Mbps. The maximal radius of a mobile's radio coverage is up to 100 meters. The transmission rate of a mobile node is 2 Mbps. The maximal speed of mobile nodes is equal to 20 m/s which simulates a car scenario. Our network has a spatial density equal to $10^{-4}$ $m^2$. The ratio of the number of mobile to immobile nodes is equal to 0.33. Only the exponentially distributed time intervals have been investigated. The simulation time is equal to 1000 s. It was assumed that the physical layer corresponded to the original 1- and 2-MHz direct sequence 802.11 physical layer which was adopted for the MAC layer. We have used the RWP model [4]. In this model, each node is at a random point at the start of the simulation and after pause (here pause is equal to 0) selects a random destination at 15 m/s for a period of time uniformly distributed between 5 and 11 seconds.

We start our simulation with the RWP mobility model. With this model, a node is allowed to move beyond the boundary of the given space. We first tested the relationship between $L(T_p)$ and the predicted link available time $T_p$. In our simulation $L(T_p)$ can approximate $\frac{\overline{T}_r}{T_p}$, where $\overline{T}_r$ is the mean time that a link will be continuously available corresponding to a prediction time $T_p$. The experimental results for the link availability $L(T_p)$ for radius equal to 100 m as a function of the predicted time $T_p$ and the speed obtained through the randomly generated values from 0 up to 20 m/s are shown in Fig. 3.

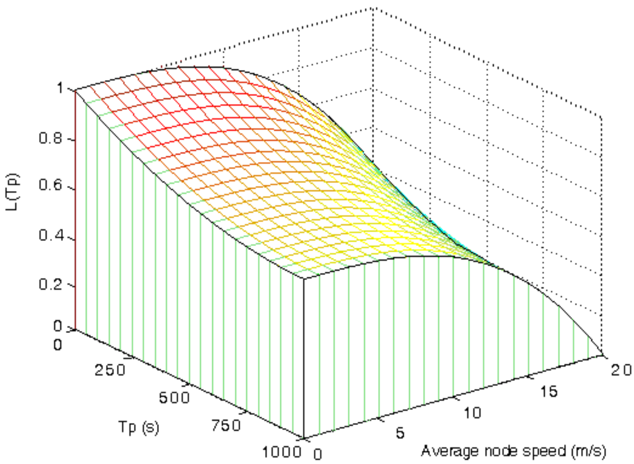



**Fig. 3.** The link availability as a function of the predicted time $T_p$ and the average speed of nodes for radius equal to 100 m

Figure 4 shows the link availability as a function of the predicted time for an average node speed equal to 15 m/s and radius equal to 50, 75, 100 meters, respectively. It seems that the radio link availability dependent on the predicted time $T_p$ can be improved by the increase of the radius of the clusterheads.

Figure 5 shows the whole network lifetime for mobile network as a function of the node transmit power, $P_1$ (identical for all nodes). It can be seen that the lifetime decreases when a value of $P_1$ increases. However, the network lifetime significantly increases as the number of the radio link grows, since the energy available in the system is better exploited.

To answer the question: how better is the algorithm for the maximization design of mobile ad hoc and sensor networks with the only aim of maximizing the minimum lifetime, we compared an algorithm from the literature [5]. In order to answer this question, we compared both solutions. In the first solution, a given number of nodes are randomly deployed. For these nodes we only calculated the lifetime of the whole network without the creation of its topology. In the second solution, by used the algorithm outlines in Fig. 1, we found all the best topologies which provide the maximum lifetime of the network. We computed the percentage gain in any of the minimum lifetimes as a function of the number of nodes defined as

$$G(n) = \frac{\max\{\tau^s_{\text{net}}(n)\} - \tau^s_{\text{net}}(n)}{\max\{\tau^s_{\text{net}}(n)\}} \times 100\%, \quad n = 1, \ldots, N \tag{16}$$

where $\tau^s_{\text{net}}(n)$ is the minimum lifetime for network, $\max\{\tau^s_{\text{net}}(n)\}$ is the maximum minimum lifetime, $n$ is the number of nodes.

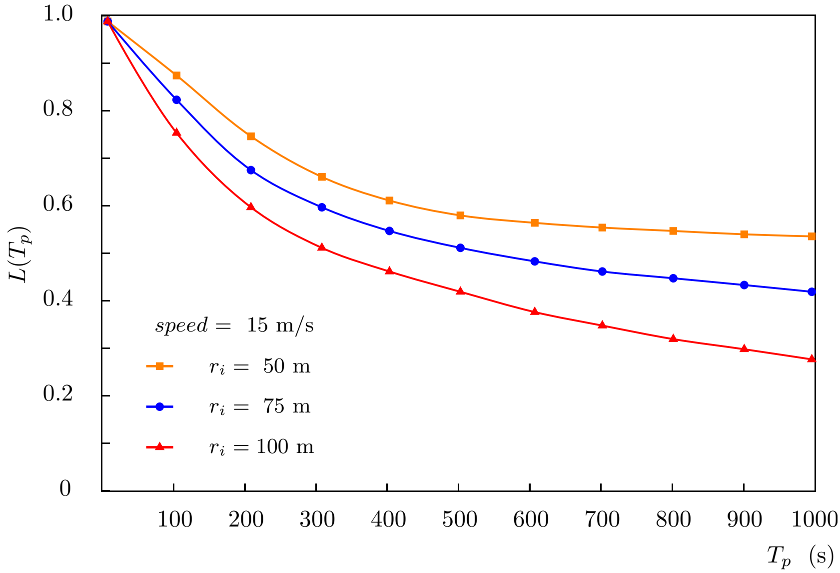



**Fig. 4.** Radio link availability as function of the predicted time $T_p$ for average node speed equal to 15 m/s

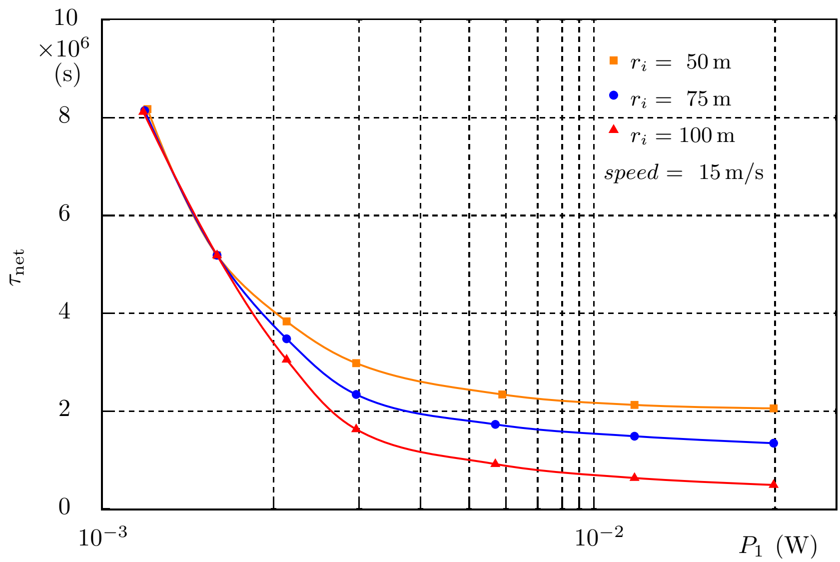



**Fig. 5.** Network lifetime of a mobile network as a function of the node transmit power $P_1$ for the average node speed equal to 15 m/s

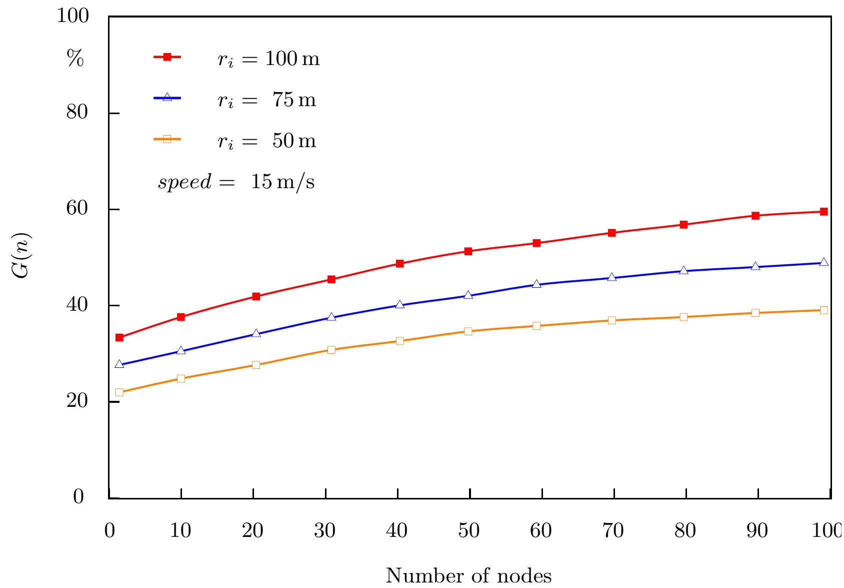



**Fig. 6.** Obtained gain in any minimum lifetime for the average node speed equal to 15 m/s as a function of the number of nodes

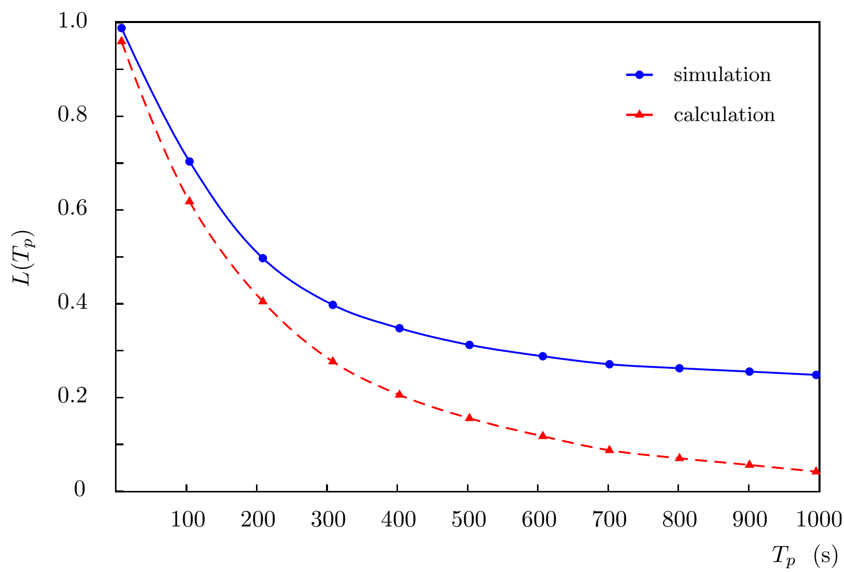



**Fig. 7.** Comparing the radio link availability from simulation and calculation for $\lambda^{-1} = 185\,\mathrm{s}$

Figure 6 shows the obtained gain in any minimum lifetime for the average speed of nodes equal to 15 m/s as a function of the number of nodes. Figure 7 shows the results.

Show that the gain of a created topology in maximizing lifetime provides ca. 35% better results in any number of nodes for the higher speed of nodes.

Finally, to validate the method of radio link availability estimation, we compared it with non-exponentially distributed epoch lengths. In this approach we used a model in which the speed was selected uniformly from 0 to 15 m/s. Thus, the epoch length is the ratio of the distance to the destination over the speed. Figure 7 shows obtained simulation results and calculation results (dashed line in the figure) in this case for the calculated parameter $\lambda^{-1} = 185$ s. As we can see from Fig. 7, the difference between both values is smaller as space size increases. Thus, the link availability estimation, which is given by Eq. (10), can be used for this case.

## 5 Conclusion

The problem of maximizing the lifetime of mobile ad hoc and wireless sensor networks belngs to the most important problems in the designing of these networks. We focused on the model which assumes the communication between the clusterhead nodes and the mobile nodes.

By the use of the radio link availability estimation, we take into consideration any cause radio links why break frequently. The given optimization and the pseudo-code of the design procedure allows us to build an energy provisioning mobile ad hoc and wireless sensor networks. One practical interest of this method is to develop a path selection metric in terms of the path reliability, which can maximize the lifetime of the whole network. However, the proposed method for all non-exponentially distributed time intervals needs further studies.

## References

1. Basagni, S., Conti, M., Giordano, S., Stojmenovic, I. (eds.): Mobile Ad Hoc Networking. IEEE Press/John Wiley and Sons, Hoboken (2004)
2. Bhardwaj, M., Garnett, T., Chandrakasan, A.: Upper Bounds on the Lifetime of Sensor Networks. In: Proc. IEEE Int. Conf. Commun., vol. 3, pp. 785–790 (2001)
3. Boukerche, A., Nikoletseas, S.: Algorithmic Design for Communication in Mobile Ad Hoc Networks. In: Calzarossa, M.C., Gelenbe, E. (eds.) MASCOTS 2003. LNCS, vol. 2965, pp. 235–253. Springer, Heidelberg (2004)
4. Broch, J., et al.: A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. In: Proc. ACM MobiCom 1998, Dallas (1998)
5. Chang, J.H., Tassiulas, L.: Maximum Lifetime Routing in Wireless Sensor Networks. IEEE Trans. on Networking 12(4), 609–619 (2004)
6. Dong, Q.: Maximizing System Lifetime in Wireless Sensor Networks. In: Proc. IEEE 4th Int. Symp. Inf. Process. Sensor Netw., pp. 13–19 (2005)
7. Ephremides, A.: Energy Concerns in Wireless Networks. IEEE Wireless Communications 9, 48–59 (2002)

8. Hu, Z., Li, B.: On the Fundamental Capacity and Lifetime Limits of Energy-constrained Wireless Sensor Networks. In: Proc. 10th IEEE Real-Time and Embedded Technology and Applications Symposium, pp. 2–9 (2004)
9. Johnson, D., Maltz, D.: Dynamic Source Routing in Ad Hoc Wireless Networks, Mobile Computing, pp. 153–181. Kluwer Academic Publishers, Boston (1996)
10. Kang, I., Poovendran, R.: Maximizing Static Network Lifetime of Wireless Ad Hoc Networks. In: Proc. IEEE Int. Conf. on Comm (ICC), vol. 3, pp. 2256–2261 (2003)
11. McDonald, A.B., Znabi, T.F.: A Path Availability Model for Wireless Ad Hoc Networks. In: Proc. IEEE WCNC, New Orlean, USA, pp. 35–40 (1999)
12. Kim II, K., Kim, S.H., Kim, S.H.: A Novel Overlay Multicast Protocol in Mobile Ad Hoc Networks Design and Evaluation. IEEE Trans. on Vehicular Technology 54(6), 2084–2101 (2005)
13. Ramachandran, B., Shanmugavel, S.: A Power Conservative Cross Layer Design for Mobile Ad Hoc Networks. Information Technology Journal 4(2), 125–131 (2005)
14. Santi, P.: Topology Control in Wireless Ad Hoc and Sensor Networks. John Wiley and Sons, Ltd., Hoboken (2005)
15. Jiang, S., He, D., Rao, J.: A Prediction-based Link Availability Estimation for Mobile Ad Hoc Networks. IEEE INFOCOM 3, 1745–1752 (2001)
16. Su, W., Gerla, M.: IPv6 Handoff in Ad Hoc Wireless Networks Using Mobility Prediction. In: Proc. IEEE GLOBECOM, pp. 271–275 (1999)
17. Toh, C.K.: Maximum Battery Lifetime Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks. IEEE Comm. Mag. 19, 138–147 (2001)
18. Wang, S.Y.: On Choosing an Appropriate Path Repair Design for Vehicle-formed Mobile Ad Hoc Networks on Highways. In: Proc. IEEE 16th Int. Symp. on Personal, Indoor and Mobile Radio Communications, vol. 2, pp. 1337–1341 (2005)
19. Xia, X., Ren, Q., Lian, Q.: Cross-layer Design for Mobile Ad Hoc Networks: Energy, Throughput and Delay-aware Approach. In: Proc. IEEE Wireless Communications and Networking Conference, pp. 770–775 (2006)

# Modeling of the Throughput Capacity of Hierarchical Mobile Ad Hoc and Sensor Networks

Jerzy Martyna

Institute of Computer Science, Jagiellonian University, ul. Łojasiewicza 6, 30-348 Cracow, Poland
martyna@softlab.ii.uj.edu.pl

**Abstract.** The throughput capacity belongs to the critical parameters for the design and evaluation of wireless mobile ad hoc and sensor networks. The achievable throughput capacity depends on the network size, traffic parameters detailes of radio interactions. In this paper, we examine these factors alone and with emphasis on such parameters as connectivity, the speed of nodes, flat and hierarchical organization of network, etc. We introduce some dependencies of the throughput capacity per node whithin a cluster, throughput capacity of clusterheads, etc. for mobile and immobile ad hoc and wireless sensor network.

## 1 Introduction

Wireless mobile ad hoc and sensor networks [11] consist of a group of mobile nodes that form temporary networks without the aid of fixed infrastructure. The communication between any two nodes depends on many parameters, such as the distance between nodes, the transmission power of the transmitter, the power of Gaussian white noise, etc.

A number of papers have been devoted to the analysis of the network capacity in terms of the achievable throughput under different system models [4], [5]. In [4], the authors proposed a two-hop transmission strategy in which the traffic is first randomly spread (first hop) accross as many relay nodes as possible, and then is delivered (second hop) as soon as any of the relaying get close to the destination. In [5], the capacity of a fixed ad hoc network was described in which the nodes' locations are fixed but randomly distributed. The authors proved that the achievable throughput between any randomly selected source-destination pair is in the order of $O(1/\sqrt{n})$, where $n$ is the number of nodes per unit as area increases.

In one of the earlier works [12] it was stated that fading actually increases the achievable rate regions (in opposition to the overall ad hoc network capacity) by providing statisticall diversity, since the best set of transmitting links can be selected. In the paper by the same authors [13] it was showed that although fading reduced the transport capacity lower bound by a logarithmic factor, it actually increased the overall network capacity.

The term defined as the transmission capacity was defined at first by S. Weber et al [14]. This measure allows to quantify the achievable rates between an

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 62–71, 2009.

arbitrary pair of nearby (i.e. single hop) transmitting and receiving nodes from an outage perspective. Finally, a number of papers [1], [2] were devoted to the problem how mobility and throughput interact in the context of time-varying channels.

In many ad hoc and wireless sensor networks is introduced a two-level hierarchy (see Fig. 1). The first layer represents all the nodes which are responsible for the data transmission to the sink. It consists only of clusterhead nodes and nodes belonging to the so called *backbone network*. The secondary layer is composed of ordinary nodes which collect the information and send it to the clusterheads. Another idea for hierarchy is to locally mark some nodes as having a special role, for instance, controlling neighboring nodes. In this sense, clusters of nodes can be formed. The "controllers" of such groups are often referred to as clustering. Many clustering scheme for mobile networks have been proposed [3], [9]. Previous research in clustering in immobile networks mainly focusses on how to form clusters with a good shape, such as minimum overlap of clusters, etc. In the clustering methods for mobile networks, the new algorithms, such as highest degree algorithm, are proposed.

The main goal of this paper is to introduce new dependencies of the throughput capacity of hierarchical mobile ad hoc and sensor networks. We characterized the network performance and obtained the new measures, which are based on defining the connectivity, the network size, the maximum number of hops required for transmission between any given pair of nodes, etc.

The paper is organized as follows. In Sect. 2 we formulate the term of the throughput capacity of ad hoc and WSNs (*Wireless Sensor Networks*). Section 3 provides the throughput capacity in immobile hierarchical ad hoc and sensor

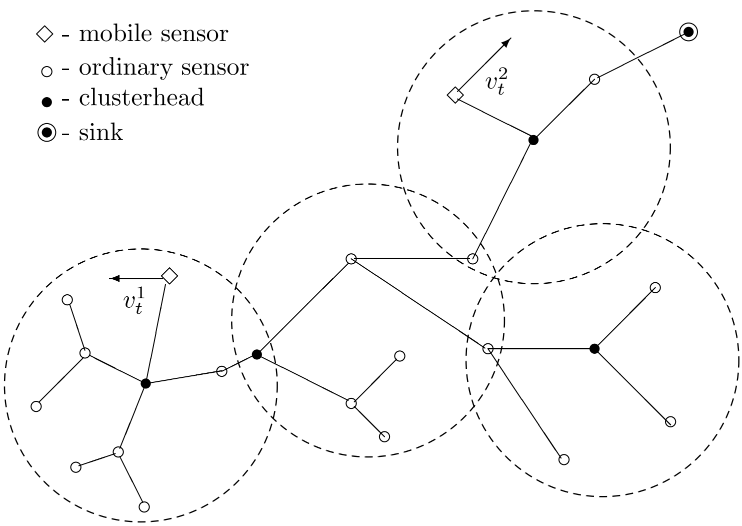



**Fig. 1.** An example of two-level hierarchy in mobile sensor network

network. In Sect. 4 we introduce the throughput capacity for mobile ad hoc and sensor networks. Section 5 presents some simulation results. Finally, some concluding remarks are stated in Sect. 6.

## 2 The Throughput Capacity of Ad Hoc and Sensor Networks in General

In this section, we give an overview of the throughput capacity of ad hoc and sensor networks.

The channel capacity or information capacity were originally defined by Shannon as the maximum information reliability transmitted from source to destination over a communication channel. The capacity is defined as

$$C = \lim_{t \to \infty} \frac{N(t)}{t} \tag{1}$$

where $N(t)$ is the maximum allowed number of signals in duration $t$. This definition concerns only the possible rate between the source and destination, neglecting the role of distance between the source and the destination.

The throughput was used as a measurement of the ability the information transmit in the network. It is the amount of data transmitted from the source to the destination during a time unit. The maximum throughput is in fact the information capacity of the channel, where the maximum value is taken for all the sources of information.

In [14], the authors formulated a new measure of the capacity, namely the *transmission capacity*. It is defined as the maximum density of successful transmissions multiplied by their data rate

$$c^{\epsilon} = \lambda^{\epsilon} b(1 - \epsilon) \tag{2}$$

where $\lambda^{\epsilon}$ denotes denotes the maximum contention density such that a fraction of $\epsilon$ of the attempted transmissions are permitted to fail, $b$ is the average rate that a successful user achieves in a bit/sec [Hz].

We note that the transmission capacity is roughly linearly scaled with the probability $\epsilon$. As shown in the paper by Weber [14], the transmission range $r$ decreases the transmission capacity according to the $O(r^{-2})$. Furthermore, the transmission range scales with $n$ – the number of nodes, as $r^2 = 1/n$. Finally, the transport capacity can be obtained by $\lambda \cdot r$, which scales as $O(\sqrt{n})$ , the same value as that given by Gupta [5].

The throughput capacity (or transport capacity) was defined by Xie [8] as the supremum of all end-to-end throughputs that are achievable in the network, namely

$$\lambda_i = \lim_{T \to \infty} \frac{b_i(T)}{T} \quad \text{for} \quad 1 \leq i \leq n \tag{3}$$

where $\{\lambda_i, R_{(i)}\}$ is the set of an active sender-receiver pair. The throughput capacity is not able to reflect the transmission ability of each node. Moreover, this definition does not take into consideration the average distance between sources and destinations.

## 3 The Throughput Capacity of Immobile Hierarchical Ad Hoc and Sensor Networks

In this section, we consider a case of the throughput capacity of immobile hierarchical ad hoc and sensor networks.

We assume for immobile hierarchical ad hoc and sensor networks that we have only two groups of nodes. To the first group belong all the nodes defined as clusterhead nodes. The clusterhead nodes in the hierarchical networks act following the principle: collect the data from ordinary nodes and create a backbone network. A backbone network is used to transmit all the collected data to the sink of the network.

The clusterhead nodes are usually equipped with multiple radios and can work parallelly. We assume that the throughput capacity of a single clusterhead node in a immobile (static) network is given by

$$T_s^{(c)} = \frac{W_1}{\sqrt{h}} \tag{4}$$

where $h$ is the number of clusterhead nodes in the network, $W_1$ is the channel bandwidth of the node.

We suppose that the throughput capacity per node within a cluster in the immobile ad hoc and sensor networks is given by

$$T_s^{(n)} = \frac{W_2}{\sqrt{n/h}} \tag{5}$$

where $n$ is the total number of nodes of an ad hoc and sensor network.

We assume that all the nodes are uniformly distributed. Thus, we can achieve the condition to obtain the optimal throughput capacity in the immobile network. The throughput capacity of each clusterhead node in a immobile ad hoc and sensor network must satisfy the condition

$$\frac{n}{h} \cdot \frac{W_2}{\sqrt{n/h}} \leq \frac{W_1}{\sqrt{h}} \tag{6}$$

The given condition means that the total accessible throughput capacity of the clusterhead node must be greater or equal to the product of the mean number of the nodes in the cluster and the throughput capacity per node within a cluster.

Thus, in a well-balanced system where the throughput capacity of a single cluster is not greater than the throughput capacity of the backbone network, we obtain the optimal number $h^\star$ under which the throughput capacity of the cluster achieves the maximum throughput capacity while still meeting inequality (Eq. 6). Both curves are plotted in Fig. 2. The optimal value of clusters is equal to the value of h where both curves intersect.

It is obvious that when $h < h^\star$, the throughput capacity of the backbone network is wasted. While $h > h^\star$, the backbone is overloaded and cannot handle all the traffic from the networks. As a result, increasing network size or input data will not increase the throughput capacity per node within cluster (see the red line in Fig. 2).

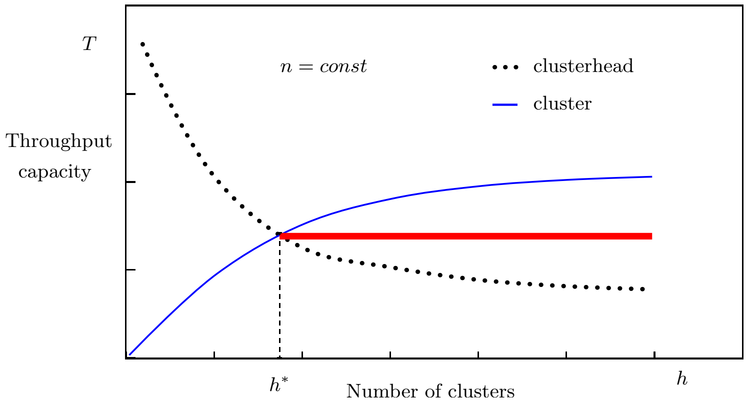



**Fig. 2.** Throughput capacity per node within a cluster and clusterhead nodes as a function of number of clusters

## 4 The Throughput Capacity of the Hierarchical Mobile Ad Hoc and the Sensor Networks

In this section, we determine the optimal number of mobile nodes in dependence of the throughput capacity of clusterhead nodes and ordinary, immobile nodes.

In this paper, we modeled node mobility by using the Random Waypoint (RWP) model. It has been introduced by Johnson and Maltz [7] to study the performance of the routing protocols. The RWP model is representative of an individual movement, obstacle-free scenario: each node moves independent of each other (individual movement), and it can potentially move in any subregion. For instance, the RWP model can arise when users move in a large room or in a open air, etc.

We assumed that the network is under the optimal circumstance. It means that the network is partitioned equally. As in the previous case, we supposed that all the nodes are uniformly distributed in the network. Additionally, we take into consideration the fact that the mobile nodes in the network can communicate only with the clusterhead nodes. It means that all the clusterhead nodes are equipped with multiple radios which use a separate bandwidth and work parallelly.

Let $n$ denote the total number of all the nodes including the clusterhead nodes and ordinary nodes. In our analysis $n$ and $m$ are constant. $m$ denotes the number of mobile nodes and variable $h$ denotes the number of clusterhead nodes. Let $W_1, W_2, W_3$ denote the channel bandwidth of the clusterhead, and ordinary and mobile nodes, respectively. Now, we can state the throughput capacity of the clusterhead node in the mobile network, namely

$$T_m^{(c)} = \frac{W_1}{\sqrt{h}} \tag{7}$$

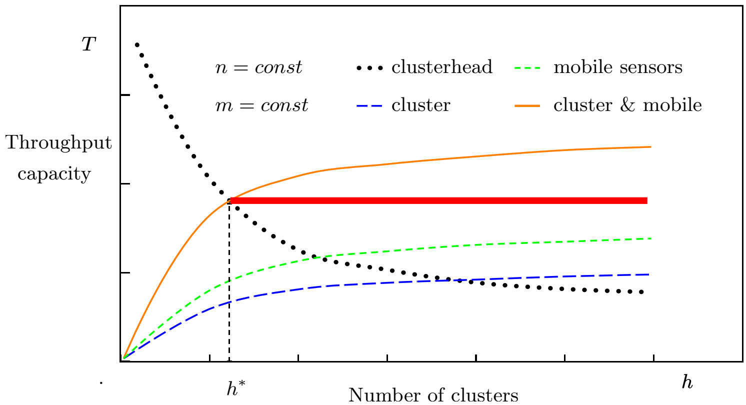



**Fig. 3.** Throughput capacity per nodes within a cluster, mobile nodes and clusterhead nodes as a function of number of clusters

The throughput capacity of a single immobile node in the mobile network is given by

$$T_m^{(n)} = \frac{W_2}{\sqrt{(n-m)/h}} \tag{8}$$

The throughput capacity of the mobile node is given by

$$T_m^{(m)} = \frac{W_3}{\sqrt{m}} \tag{9}$$

Under the optimal circumstances, if the network is partitioned equally, we assume that the average number of ordinary nodes in each cluster is equal $n/h$ and average mobile nodes in each cluster $m/h$ are given.

Since both $n$ and $m$ are fixed, all the three throughput capacities are the only functions of $h$. The throughput capacity of the backbone network must be greater or equal than the sum of the throughput capacity of mobile nodes in the cluster and the throughput capacity of nodes within the cluster, namely

$$\sum_{i=1}^{m} T_{m,i}^{(m)} + \sum_{j=1}^{n} T_{m,j}^{(n)} \leq \min_{c=1,\ldots,h} T_m^{(c)} \tag{10}$$

Now, we plot the summarized throughput capacities of mobile and ordinary nodes per cluster. We plot both curves and obtain the point of intersection. Point $h^\star$ is equal to the value of the optimal number of clusters for a given number of mobile and immobile nodes within a cluster.

Figure 3 shows that the throughput capacity of the clusterhead nodes is bound by two functions. One of them is the number of mobile nodes and the other is the number of nodes within a cluster. When $h < h^\star$, the clusterhead nodes are not congested. If $h > h^\star$, the clusterhead nodes are overloaded and unable to service all the traffic from the network. Beyond the point $h^\star$ a node in ad hoc or sensor network will not have time for successful transmission of any incoming data. Thus, the throughput capacity per node within cluster remains at constant level even the network size or the input data increase.

## 5 Numerical Experiments of the Mobile Hierarchical Ad Hoc and Sensor Networks

In order to evaluate the throughput capacity of the mobile ad hoc and sensor network, we use our model for the calculation of the main parameters of these networks.

Our models for calculation of the throughput capacity are applicable in the design of ad hoc and sensor networks. Using the given dependencies, assuming that each nodes own traffic varies between 0 and 100 kbps, we have computed (see Fig. 4) the maximum allowable throghput capacity per clusterhead node as a function of the number of nodes in static ad hoc or sensor network. As we can seen, when network increases, due to the increase in multi-hop traffic, nodes could not get rid of their own data.

Another effect visible from the computation is the maximum throughput capacity per node within a cluster as a function of the number of nodes in mobile ad hoc and sensor network (see Fig. 5). The assumed speed of mobile nodes is equal to 3 m/s which simulates a pedestrian scenario. We see when the mobility speed increases, the throughput capacity per clusterhead increases as well.

In order to verify the correctness of our method, we compared it with upper bound on output bit rate per node obtained from the Shannon channel capacity formula [10]. We recall that if we know the expected value of the ratio of the

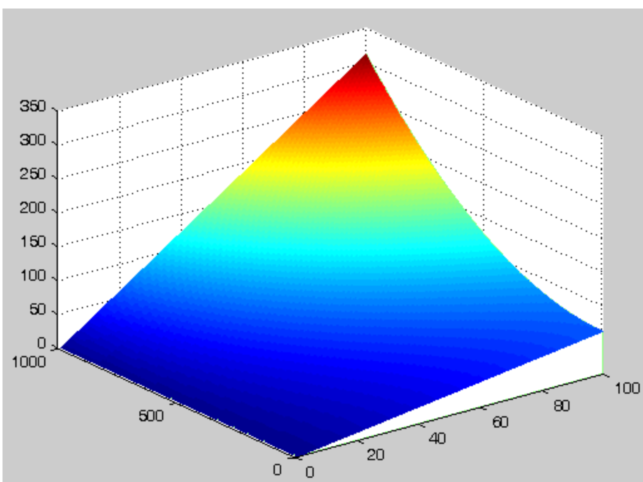



**Fig. 4.** The throughput capacity per clusterhead node as a function of the number of nodes in the static network (up to 1000) and each node's own traffic varies from 0 to 100 kbps

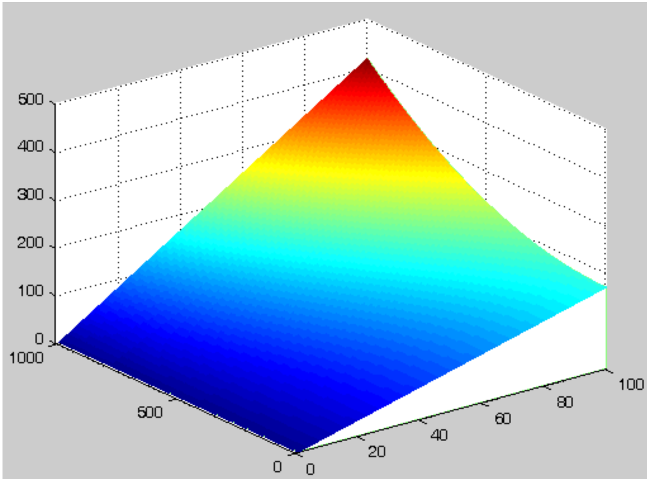



**Fig. 5.** The throughput capacity per clusterhead node as a function of the number of nodes in the mobile network (up to 1000) and each node's own traffic varies from 0 to 100 kbps. The assumed speed of mobile nodes is equal to 3 m/s.

carrier to inference ($C/I$), we can use the Shannon channel capacity formula to find an upper bound on the reliable transmission between two neighboring nodes over the radio channel, namely

$$W = B\log_2(1 + E[C/I]) \tag{11}$$

where $B$ is the channel bandwidth.

In ad hoc and sensor networks an additional restriction on the capacity is imposed by the MAC protocol. When a transmission between two nodes is established, other nodes will be prohibited from simultaneous transmission. Therefore, the capacity in the radio channel is equally divided between all nodes competing to gain access to the medium. Let $\nu/\rho$ be the fraction of the nodes that gain access to the medium at any time interval, where $\rho$ is the node density and $\nu$ is the interfering node density. Thus, the Shannon channel capacity formula in these networks indicates the maximum output bit rate, $R_{\text{max\_out}}$, per node that can be supported by the network. Using the Eq. (11) we find:

$$R_{\text{max\_out}} = \frac{\nu}{\rho}W = \frac{\nu}{\rho}B\log_2(1 + E[C/I]) \tag{12}$$

According to the so-called honey-grid model, used for modeling ad hoc and sensor networks, $R_{\text{max\_out}}$ can be expressed formally as [6]:

$$R_{\text{max\_out}} = \frac{B}{1+3a(a+1)}\log_2\left(1 + \frac{(a+1)^{\eta-1} g \sum_{j=1}^{a} j^{-(\eta-1)}}{3a(1-e^{-\lambda E[h]})\sum_{j=1}^{\lfloor \frac{k}{a+1} \rfloor} j - (\eta - 1)}\right) \tag{13}$$

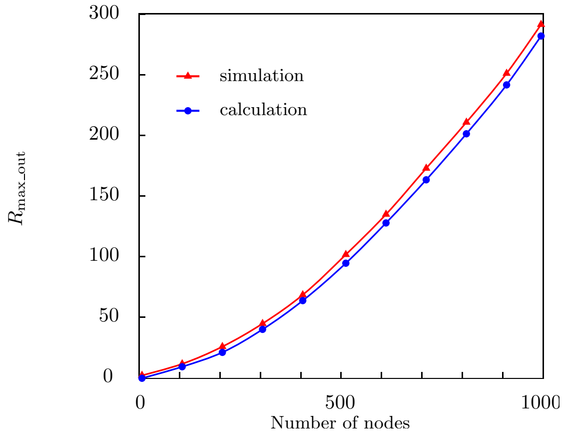



**Fig. 6.** Comparing the output bit rate per node from analytical model and simulation

where $a$ is the reach of a node in the center of the configuration, $g$ is the processing gain, $E[h]$ is the average number of hops, $\lambda$ is the mean arrival rate of new packets per node per time-slot (node's own traffic), $k$ is the size of the network expressed in terms of co-centered hexagonal rings, $\eta$ is pathloss exponent.

We compared the maximum output bit rate given by Eq. (13) and the maximum output bit rate per node provided by our model with simulations. Figure 6 shows both results. There is very good agreement between the analytical model obtained from the Shannon channel capacity formula and simulation results.

## 6 Conclusion

In this paper, we have studied the problem of the throughput capacity of mobile hierarchical ad hoc and WSNs networks. We assumed that nodes in a network are not always mobile, sometimes they are immobile. It allows us to compare the throughput capacity of mobile hierarchical ad hoc and sensor networks with immobile networks. We obtained results that describe the dependencies between the number of mobile nodes, the number of clusterhead nodes and the ordinary nodes in the network. We established that under the optimal circumstances, if the network is partitioned equally, the optimal number of mobile and immobile nodes can be determined.

We obtained the conditions for the throughput capacity in the hierarchical mobile ad hoc and sensor networks.

## References

1. Bansal, N., Liu, Z.: Capacity, Delay and Mobility in Wireless Ad Hoc Networks. In: Proc. of the IEEE INFOCOM 2003, pp. 1553–1563 (2003)
2. Gamal, A.E., Hammen, J., Prabhakar, B., Shah, D.: Throughput-Delay Trade-off in Wireless Networks. In: Proc. of the IEEE INFOCOM 2004, pp. 464–475 (2004)

3. Gerla, M., Tsai, J.T.: Multicluster, Mobile, Multimedia Radio Network. ACM Baltzer Journal of Wireless Networks 1(3), 255–265 (1995)
4. Glossglauser, M., Tse, D.: Mobility Increases the Capacity of Ad Hoc Wireless Networks. In: Proc. IEEE INFOCOM 2001, pp. 1360–1369 (2001)
5. Gupta, P., Kumar, P.R.: The Capacity of Wireless Networks. IEEE Trans on Information Theory 46(2), 388–404 (2000)
6. Hekmat, R., Van Mieghem, P.: Interference in Wireless Multi-hop Ad Hoc Networks and Its Effect on Network Capacity. Wireless Networks Journal 10, 389–399 (2004)
7. Johnson, D., Maltz, D.: Dynamic Source Routing in Ad Hoc Wireless Networks. In: Mobile Computing, pp. 153–181. Kluwer Academic Publishers, Dordrecht (1996)
8. Xie, L.-L., Kumar, P.R.: A Network Information Theory for Wireless Communication, Scaling Laws and Optimal Operation. IEEE Trans. on Information Theory 50(5), 748–767 (2004)
9. Lin, C.R., Gerla, M.: Adaptive Clustering for Mobile Networks. IEEE Trans. on Selected Areas in Communications 15(7), 1265–1275 (1997)
10. Peterson, L.L., Davie, B.S.: Computer Networks: A Systems Approach, pp. 94–95. Morgan Kaufmann, San Francisco (1996)
11. Shorey, R., Ananda, A., Chan, M.C., Ooi, W.T. (eds.): Mobile, Wireless, and Sensor Networks. John Wiley and Sons/IEEE Press, Ltd., Hoboken (2006)
12. Toumpis, S., Goldsmith, A.J.: Capacity Regions for Wireless Ad Hoc Networks. IEEE Trans. on Wireless Communications 24(5), 716–748 (2003)
13. Toumpis, S., Goldsmith, A.J.: Large Wireless Networks Under Fading, Mobility and Delay Constraints. In: Proc. of the IEEE INFOCOM 2004, pp. 609–619 (2004)
14. Weber, S.P., Andrews, J.G., Yang, X., de Veciana, G.: Transmission Capacity of Wireless Ad Hoc Networks with Outage Constraints. IEEE Trans. on Information Theory 5(12) (2005)

# Object Oriented Vertical Communication in Distributed Industrial Systems

Rafał Cupek[1], Marcin Fojcik[2], and Olav Sande[3]

[1] Silesian University of Technology, Institute of Informatics
rcupek@polsl.pl
[2] Sogn og Fjordane University College
marcin.fojcik@hisf.no
[3] Cronus Automation Vest AS
olav.sande@cronus.no

**Abstract.** Low level industrial systems use simple data representation which reflects flat input and output structure of control industrial system. This structure is directly mapped to the flat memory allocation. The structural data organization is used in upper layers of industrial computer system. This article contains compare analysis of tag and object oriented data model used for vertical communication in distributed industrial computer systems. This analysis is based on classical tag approach used in the most of classical visualization systems and object oriented approach introduced in OPC UA architecture.

## 1 Vertical Communication in Distributed Industrial Systems

In the past, the structure of production systems was centrally orientated. This means, we had one control device and all field devices transferred their data through the communication systems. With the development of microelectronic technology in the last years, industrial informatics system became more distributed from the topological point of view and from system logic as well.

Modern control systems can be divided into layers and the most of classifications define four layers for control, visualization and production support systems:

- Field Layer with instrumentation.
- Control Layer with automation devices.
- Real Time HMI (Human Machine Interface) Layer with visualisation devices.
- Real Time MES (Manufacturing Execution System) Layer with data processing devices.

Above these, the ERP (Enterprise Resource Planning) Layer could be defined as a separate layer.

Suppliers of proprietary control systems tend to hide the layered nature of their products, but even these systems are in fact layered. For open systems it is important to define these layers, and then it is possible to use a variety of

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 72–78, 2009.

Control Layer Devices and a variety of HMI Layer solutions, given that there is a uniform way of communicating between the devices.

There are some norms dedicated for low level industrial control systems which describe different standards [1,5] in industry: ISO15926 (elements, communication), IEC/EN 61131 (programming of PLC), IEC 61850 (data models, grouping of data, commands, types of transfer), IEC 61158 (Digital Data Communication for Measurements and Control – Fieldbus in Industrial Control Systems). Each of them describes issues related to given layer. None of them defines the inter-layer communication relations.

From another point of view there are some standards like ISA 95 or IEC/ISO 62264 dedicated for MES layer which use the object oriented system model and define explicit needs for object oriented input/output data format. This situation causes many problems in practical realization of multi-layer, large scale industrial computer systems.

Because of many existing systems which do not support object oriented approach an idea of creating a "middleman" in the vertical data exchange between low layer and upper layers of industrial computer system was introduced. There are some factory middleman solutions. One of the object oriented communication solutions' is PROFINET CBA network. Automation system realized on PROFINET CBA base enables creation of large size component based automation systems made from parts delivered by different producers. However PROFINET CBA communication standard may be used in large scale distributed industrial systems but the underlying DCOM communication protocol may be the reason for low popularity of this communication standard [7].

The most common used "middleman" solution is developed by OPC foundation OPC client – server industrial communication protocol which became the facto standard for vertical communication in industrial computer systems.

During OPC standard evolution many of system aspect have to be changed. There were new services and new OPC interfaces introduced, but the primary data representation as a VTQ (Value Time Quality) structure was unchanged in next OPC Data Access, Alarm and Events or Historical Data Access versions. The main OPC communication problems were platform dependence on Microsoft DCOM technology, and difficulty of implementation inside of low level industrial devices. The new approach for data representation was used in OPC Unified Architecture [2]. This chapter concerns on structure of data exchange between low and upper layers of industrial computer system on the base of tag and object oriented approach.

## 2 Tag Oriented Approach

The PLC architecture is based on flat input and output data space. Physical signals are grouped to fit real modules to which they are physically connected. The representation of physical wiring is mapped into PLCs' memory map – also flat structure. Although dependences between signals have to be consider during control algorithm implementation they have no representation in PLC

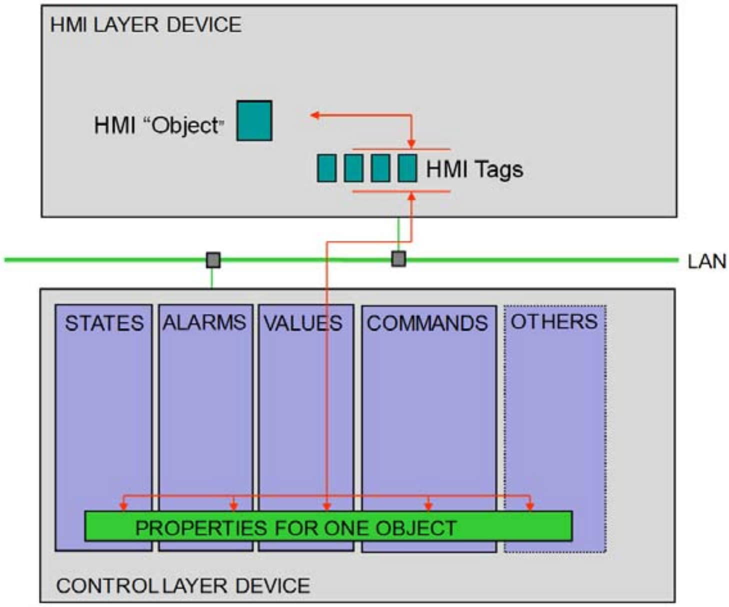



**Fig. 1.** Tag oriented system architecture

data structure. Such situation caused that communication between control layer and visualization systems has also started on the base of flat "tag oriented" data representation (Fig. 1).

A Tag is an abstract representation of process variable used in visualization system. As usual tags are directly mapped to given bits or registers addressed inside devices' memory. A TAG based system, uses polling as a communication scheme and usually has the following properties:

– Data must be organized in "arrays" to be efficient. Even so the time response of such systems is often not acceptable.
– Time stamping is only possible by the HMI Layer Devices. This implies poor time resolution.
– Information must be exposed as "public data", thus data can be "destructed" by ill managed programs.
– TA large amount of TAGs must be managed by "engineering tools", often for small changes also.
– The system bogs down network and Control Layer Devices with "meaningless" work. Data rarely changes during normal plant operation.

As Programmable Controllers and HMI systems tends to enforce some sort of object orientation scheme it is imperative that communication between this two systems is also performed in an object oriented manner. That is, communication should be a peer to peer communication between object components and not through TAGs and TAG arrays. That object oriented approach is not supported in classical solutions of low level industrial networks.

One of the most advanced examples of vertical communication systems is Wonderware System Platform – called ArchestrA technology [6]. The input / output communication servers are introduced for vertical data exchange. ArchestrA architecture allows not use only simple binary, analog or text variables reflecting to direct real input and output representation but more composed tag structural representations called as "super tag" may be used. These "super tags" reflect to real dependences between object signals and allow to create multilevel, hierarchical structure data representation. Despite of many implementation limitations such structure may by compared to the folder organization structure rather than to object oriented approaches.

## 3 Object Oriented Approach in OPC UA

In an object oriented system data are normally stored in non-contiguous memory areas and object data are hidden in the objects. In an object based system the object components should communicate on a peer to peer basis not as simple value exchange, but as object properties and procedures which present required aspects of given data structure. Object attributes are "pushed" between the HMI Layer Device and Control Layer Device in the form of short messages, and the communication services are changed from cyclical pulling model into client-server services. The object oriented approach is shown on Fig. 2.

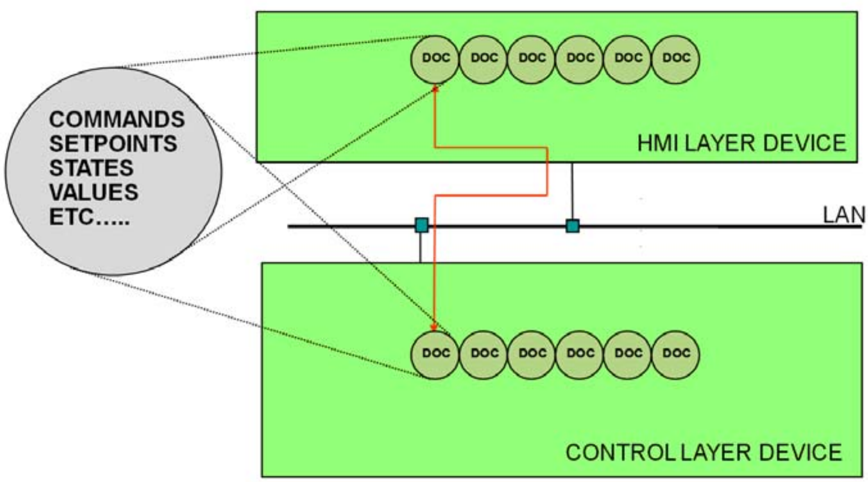



**Fig. 2.** Object oriented system architecture

An object based system usually has the following properties:

- Data does not have to be organized in “arrays” to be efficient. The time re-sponse of such systems is extremely fast.
- Information can be hidden in the objects, thus data can not be “destructed” by ill managed user programs.
- There are no TAGs to be managed by “engineering tools”. Small changes can even be performed “per hand” with great integrity. There is only one link be-tween HMI Layer Device Distributed Object Components and Control Layer Device Distributed Object Components to be managed.
- The system opens up for local data Time Stamping and Quality Code generations, even by the Control Layer Device IO sub-system (if the Control Layer Device supports such functionality).

In order for a HMI Layer Device and Control Layer Device systems to communicate in object oriented mode they either have to implement an object oriented attribute message passing scheme as a default, which is seldom the case, or it must be possible to transfer “blocks” with raw data between the two systems and implement necessary program(s) to encode and decode these raw data. The raw data represents messages intended to manipulate or reflect object attributes (i.e. commands, alarms, states, setpoints, values etc.).

HMI system of course can implement such functionality using “powerful” programming tools, thus it is possible to implement a message passing scheme for these systems. From a Programmable Controller point of view there are few truly object oriented HMI systems available. Systems claiming object orientation are normally merely “tag grouping” systems. All TAGs still has to map a “discrete” Programmable Controller resource. Currently there are no such systems available.

These were more reasons to find more advanced version of OPC – UA Unified Architecture. The market required the OPC foundation to provide better integration of alarms in the address space of a Data Access Server, especially for the process industry and building automation. To date three different OPC servers – Data Access, Alarms and Events, and Historical Data Access – with different semantics have been required, for example to capture the current value of a temperature sensor, an event resulting from a temperature threshold violation, and the historic mean temperature.

OPC UA enables all three types of data to be accessed by a single OPC server. As a result UA unifies the current DA, AE, HDA models and additional program calls into a single integrated address space [3,4]. This unified architecture reduces the number of OPC components to be installed and simplifies configuration and installation. UA also provides the additional option of storing a more detailed description of a data point directly in the OPC UA object, e.g. unit, scale factor etc. In this way OPC UA makes data management simpler, more centralized, and richer in additional information. This idea of OPC Unified Architecture is presented on Fig. 3.

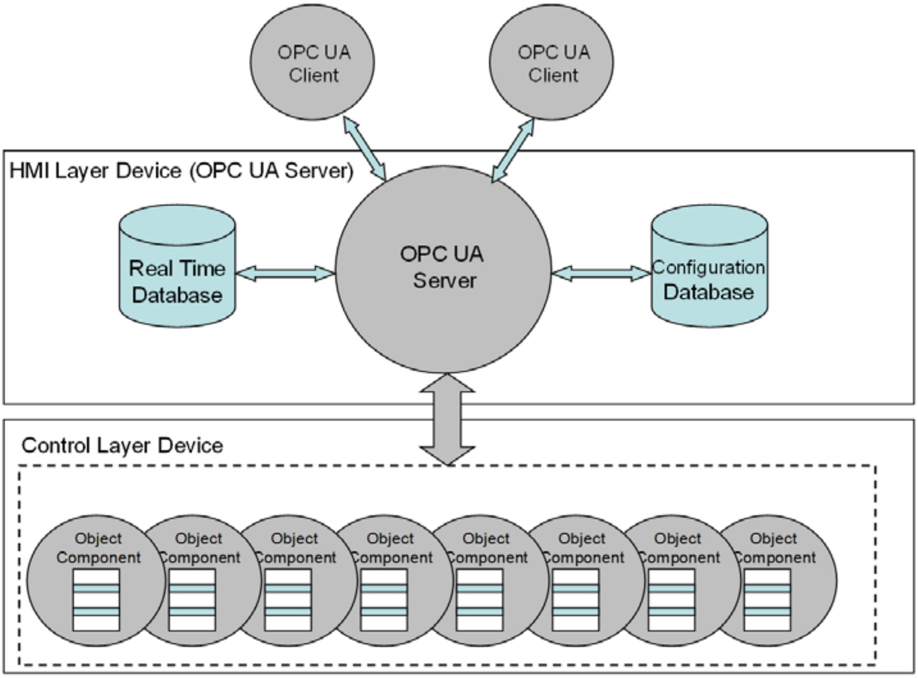



**Fig. 3.** OPC UA system architecture

New idea forced new addressing model introduced in OPC UA. OPC UA server gives access to three kinds of objects managed by OPC UA servers and available to OPC UA clients:

Type definition – because of the possibility to create object oriented data structures their clients need the possibility to discover the managed by servers object data types. In tag oriented solutions there were sets of basic data types allowed by given standard, so there was no need for additional type definition. During tag declaration standard data types has been chosen. In "super tag" data structures there was the need to define the data structure components but the user of given data type was responsible for knowledge of its structure. OPC UA data typing model do not require previous data type knowledge on client side. Client can use object oriented data types to discover their definitions stored on server. The references and cross server references are possible in type definition. Types can organize data into structures but also another object oriented mechanisms like subtyping, type inheritance are available.Types can be used for objects, variables, data and reference definitions.

Objects definition – objects reflect real system structure presented in the way defined by given type objects with their variables, properties and methods. Variables are used for presentation real process' signals which change during process execution, properties are used for object description, and in case when more complicated activity on the object is needed methods are used. Of course there is still possible to create simple object and use them in similar way like in tag oriented approach but more flexible is to use context oriented object presentation with usage of object oriented approach. The object oriented structure is also supported by events mechanisms which cruses that there is no longer need for cyclic data pulling and continuous object state checking. The information is automatically sent to the client which subscribe given event and receive a required message in the case of its evidence.

View mechanism – the large sets of data managed by OPC UA servers can be presented by the more client convenient way with the views usage. Views help to organize large data structures to present information important in the given context. Views can also use references and can reference objects on another servers.

## 4 Conclusions

Idea of using object oriented vertical communication gives some new possibilities. One of the biggest is unifying all the data in one memory model. Instead of sending two or three transmissions (value, value changing, alarm, event) it is enough to send one object with all parameters. In this way a client receive consistent data prepared in convenient format on the server's side. Next thing is that model make possible to get better resolution of Time Stamping. It can be realised in changing all cyclical exchanges into client-server model (Service Oriented Architecture). Another advantage of such communication model is the data transfer reduction on an underlying communication network also.

It is sure interesting how these new properties of OPC UA affect time parameters and requirements. Unfortunately, in the time when this article was written any OPC UA software fully compatible with specification (both client and server) wasn't commercial available. So it seems to be necessary to make theoretical estimate time requirement for OPC UA.

## References

1. ISO 15926, IEC/EN 61131, IEC 61850, ISA-95 Standards
2. Leitner, S.H., Mahnke, W.: OPC UA – Service-oriented Architecture for Industrial Applications. ABB Corporate Research Center
3. OPC Foundation: OPC Unified Architecture Specification Part 5: Information Model v.1 (2006)
4. OPC Foundation: OPC Unified Architecture Specification Part 3: Address Space Model v.1 (2006)
5. Koronios, A., Nastasie, D., Chanana, V., Haider, A.: Integration Through Standards – An Overview Of International Standards For Engineering Asset Management. In: Fourth International Conference on Condition Monitoring, Harrogate, United Kingdom, June 11-14 (2007)
6. ArchestrA IDE User's Guide, http://www.wonderware.com/support/
7. Bregulla, M., Cupek, R., Fojcik, M.: PROFINET CBA via Internet. In: Contemporary Aspects of Computer Networks, WKŁ, Warszawa, vol. 2 (2008)

# Combining Timed Colored Petri Nets and Real TCP Implementation to Reliably Simulate Distributed Applications

Wojciech Rząsa*

Rzeszow University of Technology, Wincentego Pola 2, 35-959 Rzeszów, Poland
wrzasa@prz-rzeszow.pl

**Abstract.** In recent years the importance of the problem of efficiency of distributed applications grows since the distributed computer systems become more and more pervasive. The problem is the more difficult since it connects aspects of concurrent processing and communication between different parts of applications over the Internet links with parameters changing in time. In this paper we describe the problem of estimating data transmission time over the network. The results are part of the work heading towards enabling reliable simulation of distributed applications. The model of application and resources is created using Timed Colored Petri Nets (TCPN) and it is combined with Ns-2 implementation of Linux TCP. Thus reliable simulation of concurrent processes and resource competition is ensured by the formalism of TCPN and network transmission time is estimated by implementation including real TCP code. We present preliminary results of comparison between simulations and experiments.

## 1 Introduction

Pervasive access to world wide links of the Internet and growing number of mature technologies form possibility of creating more complicated distributed applications covering subsequent areas of our activity. Technologies emerging around the Grid concept [6] are exceptionally popular in recent years and the first motivation for this work comes from this environment [16]. However the results can be useful also for the other distributed environments.

In the face of growing number of distributed systems and their applications the problem of forecasting efficiency of the designs becomes the more important. It is even more significant for the sake of complexity of the problem. Efficiency of whole application depends on efficiency of subsequent distributed elements, efficiency of communication and dependencies between concurrent activities and synchronization of the processes.

In this paper we describe a method of estimating TCP transmission time developed to enable reliable simulation of distributed applications. The overall goal of the research is to provide developers of distributed applications with a

* The author is grateful to Dr Marian Bubak for all valuable suggestions and to Mr Dariusz Rzońca for his remarks.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 79–86, 2009.

method capable of estimating efficiency of their designs even before laborious implementation. The concept of simulation method assumes that application developer provides a *High-level* model of his application and exploited resources. Then this model is automatically transformed to a formalism enabling reliable simulation. Necessarily the simulation requires a method to accurately estimate network transmission time and this problem is in the center of this paper.

## 2 Simulation Model

In this section we describe the *High-level model* of application and resources (Fig. 1) we have developed.

The resources are represented in the model as a set of *nodes* and *network segments*. For each node we define CPU(s). The network segments are currently characterized by available bandwidth, delay and additional capacity representing possible router queues. In order to indicate which network segments should be exploited by communication between particular nodes the model includes definition of *network links*. The network link has two endpoints (at the nodes) and it contains a sequence of identifiers of network segments between these endpoints. Thus it is possible to model dependencies between network traffic generated by some nodes and network bandwidth available for communication between the other nodes. Thus the network topology can be to some extent implemented in the model. However this information should be limited to the essentials only.

In our model the application is described as a set of *elements*. An *element* is a part of application which is running on one node. It can be one process or numerous processes communicating by the means provided by operating system. In either case we consider element as a whole. We do not assume any

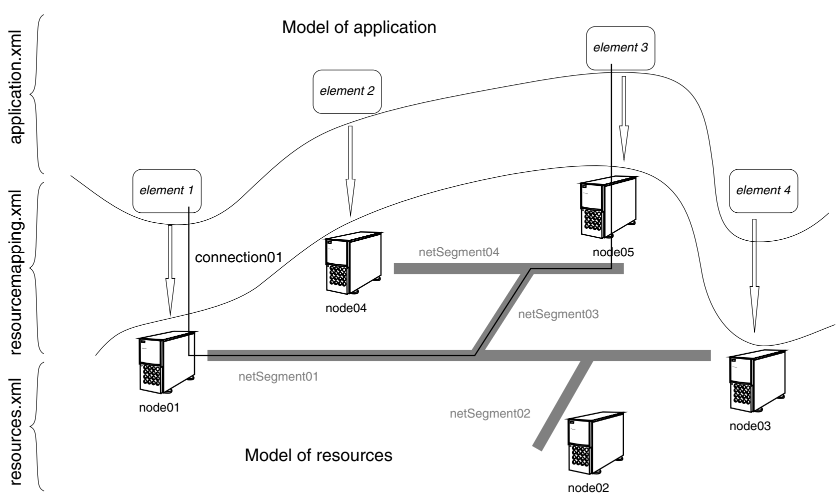



**Fig. 1.** High-level model of a distributed application and resources. Subsequent parts of the model are described in separate XML files, thus it is possible to consider one model of application in different environments.

specific programming model. An element can be a Web Service as well as Common Component Architecture component or a process exploiting Globus Toolkit communication library.

The application logic realized by each element is described in the model in terms of resource usage defined separately for each type of resource. The resource usage by the application elements is defined by expressions describing mathematical relationship between available variables (e.g. simulation time, volume of received data, etc.) and load of particular resources, e.g:

```
if ($time < 1000) { 2*$incomingVolume }
else { pow($incomingVolume,3) }
```

## 3 Enabling Reliable Simulation

Reliable simulation of a distributed application requires a method of reflecting numerous activities of concurrent processing. As a formalism enabling the analysis we have chosen Petri nets [11] that were designed to enable modeling of concurrency. The PN can be used as a graphical presentation of the activity of a system and also as a mathematical model which enables theoretical considerations. In our application we exploit Timed Colored Petri Nets (TCPN) proposed by Kurt Jensen [8]. The *High-level* model provided by simulator user is automatically transformed to TCPN based *Low-level* model and the low level model is analyzed by means of simulation. More details concerning simulator concept can be found in [15].

## 4 The TCP Problem

The concurrency is not the only problem that should be considered while analyzing distributed applications. The second major issue is a reliable assessment of network transmission time. Nowadays majority of applications communicating over the Internet exploit Transmission Control Protocol for data transfers, therefore data transmission time over the TCP connections should be reliably estimated by the simulator.

The TCP is however a significantly complicated protocol. It was first defined by [17]. Some aspects concerning network congestion management and retransmissions were precised by [1] and [2]. The first concepts were extended e.g. by [10] and other specifications over the years. Currently there is no single specification determining all details of the protocol and numerous issues are solved by designers of specific stacks. Consequently various implementations significantly differ between each other. There are also numerous approaches to sender side flow control algorithms that are crucial for transmission efficiency. Therefore the TCP behavior depends on numerous factors including network parameters (bandwidth, delay), connection state (slow start, congestion avoidance) thus previous transmissions, volume of transmitted data and finally the protocol implementation, parameters and flow control algorithm used.

## 5 Related Work

There are numerous works concerning TCP modeling, enabling estimation of its efficiency on the basis of different parameters. There are both: analytical models, and simulation models. For the sake of complexity of the problem the analytical models (e.g. [5], [4], [9]) necessarily concentrate on individual aspects of TCP protocol, assuming different simplifications concerning the other issues. For instance it is frequent to assume that data are sent incessantly and also that receivers are greedy and accept all incoming data immediately (see: [9]). It results in assumption that TCP always advertises full receive window. This is not true if sender or receiver performs processing of received data and for some time is not available for data transfers. Some works concern TCP connections used to transfer specific traffic e.g. [5]. As a consequence these models are not appropriate for general purpose distributed application simulator, where communication channels can be loaded with traffic of different characteristics, can be established over networks with different parameters and the application is assumed to perform processing of transmitted data, not only the transmissions.

We have also found works concerning simulation based analysis of TCP. The most popular network simulation tool is Ns-2 [12] (e.g. [7]), however the other simulation solutions are also exploited, including Jensen's Colored Petri Nets (e.g.: [3], [14]). The works frequently concern different versions of TCP and concentrate on comparison of some specific solutions. However they are still far from real implementations that combine the details of TCP specifications it their peculiar way.

It seems obvious that the best solution to analyze the real TCP behavior in most accurate way is to use real TCP implementation. If we could at least semi-automatically adapt e.g. Linux kernel code to our simulator it would be easy to upgrade it do future changes and the results of the simulation could be accurate. This is also not a novel idea. However extracting TCP implementation from Linux kernel encounters numerous significant problems. One difficulty is caused by the fact that Linux TCP extensively exploits kernel infrastructure in the form of functions, defines and constants. This results in dependency tree including large part of the kernel. Moreover the Linux TCP is implemented with an assumption that it is the only running instance of TCP implementation on a system and thus it is using large number of global variables. More detailed discussion of the problems and solutions can be found in [18].

The goal of [18] was to include Linux TCP into popular Ns-2 network simulator, to enable analysis of the specific implementation. Their solution was to create Linux-like implementation of TCP for Ns-2 and then adapt real congestion avoidance (CA) code from Linux to Ns-2. The interface for CA is well defined in Linux thus it is possible to easily upgrade the Linux Ns-2 TCP using newer kernels. Since the CA is crucial for TCP efficiency and the Linux-like TCP implementation can be suited to real Linux TCP this solution is supposed to give good results.

## 6 Estimation of TCP Transmission Time

At the very beginning we were looking for an analytical model of TCP performance. This solution could be naturally included in the Timed Colored Petri Net. However the analytical models appeared to be too specific for general purpose simulator. Thereafter we tried to including more complicated model of real TCP in our TCPN based model. Similar model was already described in [3]. This solution promise good results in less specific environments. However it makes difficult and sometimes impossible to reflect specifics and changes of real implementations. Therefore it can be well suited to some theoretical solutions but it will be difficult to adapt it to real implementations and future changes. However we have created TCP model in Timed Colored Petri Nets and results of simulation presented in [15] were obtained with this model.

It seems that solution presented in [18] is most flexible for future adaptations and promises most accurate results. Thus we decided to extract Linux TCP implementation from Ns-2 and include it into our simulator as a shared object with well defined interface. Thus our simulator is easily upgradable for any future TCP implementations and there is no need for changes in TCPN based model responsible for reliable modeling of concurrency and resource exploitation. This way we take the best advantages of Timed Colored Petri Nets in the area where it best fits and connect it with reliable TCP implementation which is as close to the real one as it is possible. The solution is also flexible and easy to adapt to the other TCP implementations.

The actual architecture of the combined TCPN and Ns-2 based model is presented on Fig. 2. The application model is implemented in Petri nets, when it sends data the package is passed to Ns-2 TCP send layer. The segments produced by Ns-2 TCP are passed as tokens to the TCPN model of network thus it is possible to reliably simulate network sharing between different connections. When segments are transmitted over the network they are again passed to the Ns-2 TCP receiver side. The Ns-2 performs TCP actions and finally passes data to the TCPN model of application.

The simulator of Timed Colored Petri Nets is implemented in Java, taking advantages of Petri Net Kernel and Petri Net Cube [13] packages. The Ns-2 TCP implementation is compiled to shared library and accessed from the simulator by

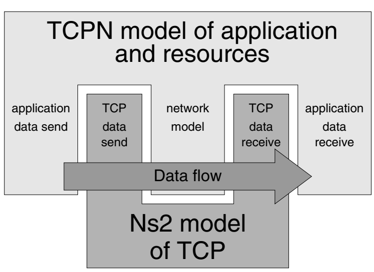



**Fig. 2.** Simulated data flow over combined TCPN and Ns-2 TCP network layer model

the means of Java Native Interface. Each connection is represented as separate Java object thus numerous connections can be independently handled by a single simulator. Timed Colored Petri Net simulator timer and global Ns-2 clock are synchronized during the simulation.

## 7 Results of Experiments and Simulations

In order to verify correctness of the solution we have performed experiments and we compare their results to the results of simulation. These preliminary tests were taken in laboratory by the use of separated network. Therefore we are able to draw conclusions concerning accuracy and actual network parameters.

The generic configuration of experiment is presented on Fig. 3. The nodes were running Linux 2.6 with BIC TCP enabled and the bit rates for subsequent network interfaces were limited by Token Bucket Filter. Similar configuration was implemented in the simulator.

The experiments consisted on transmission of ten packages of data as presented on Fig. 3. We have measured round trip time of whole transmission on `node1`. All bit rates were set to the same value of 1 Mbps for the first experiment and 2 Mbps for the second one. The application processes did not perform any computations. The results are presented on Fig. 4.

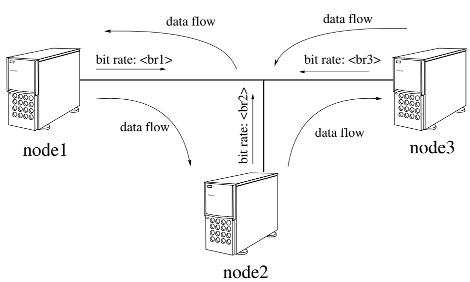



**Fig. 3.** Configuration of the experiments

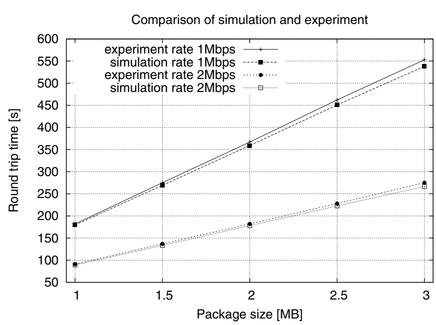



**Fig. 4.** Results of the experiments and simulations

## 8 Conclusions

Presented results prove correct cooperation between both: TCPN and Ns-2 based parts of the model. The events are correctly passed between the two environments and the clocks are properly synchronized. The network TCPN model correctly reflects network sharing between different connections. Thus we can conclude that the combination of both models is done properly. Accuracy of the results show that the Ns-2 based TCP is a good model of real TCP implementation in the examined case.

Obtained results present simple linear relationships between time and size of transmited data what is natural in this environment. However we should realize that the dependency may be significantly more complicated in more complex conditions. Especially when the application elements perform computations and cannot immediately receive data from the network or delay-bandwidth product for a network link is large. Even if the reliationships between data volume and transmission time are linear in more complex cases, the actual parameters of the function depend on large number of factors connected with network state and behavior of the application. Therefore number of simplifications were introduced in papers describing analytical analysis of TCP referenced from Sect. 5.

## 9 Summary and Future Work

Preliminary results of the simple experiments presented in this paper prove usefulness of combined TCPN and Ns-2 model of distributed applications, however further experiments should be performed in order to verify the simulator in different conditions.

Certainly accuracy of the results depends on accuracy of the Ns-2 TCP implementation, however we belive that the solution for estimation of TCP transmission time we have choosen promises as good precision as it is possible. Authors of [18] discuss results for different configurations of their TCP. In some examples the Linux TCP in Ns-2 presents good convergence with real implementation, however there are configurations in which the results are inaccurate. Moreover even for the best cases there are differences between real Linux TCP and the Ns-2 implementation that may render inaccurate results for some network or traffic parameters. However new versions of Linux TCP for Ns-2 can be easily adapted to our simulator. We should also remember, that the network transmission is not the only factor influencing results of the simulation, therefore minor inaccuracies in estimation of network transmission time can be irrelevant when we consider data processing by simulated application.

## References

1. Allman, M., Paxson, V.: TCP Congestion Control. RFC 2581 (April 1999)
2. Allman, M., Paxson, V.: Computing TCP's Retransmission Timer. RFC 2988 (November 2000)

3. De Figueiredo, J.C.A., Kristensen, L.M.: Using Coloured Petri nets to investigate behavioural and performance issues of TCP protocols. Department of Computer Science, Aarhus University, 21–40 (1999)
4. De Vendictis, A., Baiocchi, A., Monticelli, A.: Simple Models and their Limits for TCP/IP Network Analysis and Dimensioning. In: Proc. of IEEE ICC 2002 (April 2002)
5. Elteto, T., Vaderna, P., Molnar, S.: Performance Analysis of TCP Networks Loaded by Web Traffic. In: 18th International Teletraffic Congress, ITC18, Berlin, Germany, August 31 - September 5 (2003)
6. Foster, I., Kesselman, C., Tuecke, S.: The Anatomy of the Grid. International Journal of Supercomputer Applications 15(3) (2001)
7. Fall, K., Floyd, S.: Simulation-based comparisons of tahoe, reno, and sack tcp. Computer Communication Review 26, 5–21 (1996)
8. Jensen, K.: Coloured Petri Nets. In: Basic Concepts, Analysis Methods and Practical Use, Basic Concepts. EATCS Monographs on Theoretical Computer Science. Springer, Heidelberg (1994)
9. Kalampoukas, L., Varma, A., Ramakrishnan, K. K.: Two-Way TCP Traffic Over Rate Controlled Channels: Effects and Analysis, University of California at Santa Cruz, Santa Cruz, CA, USA (1997)
10. Mathis, M., Mahdavi, J., Floyd, S.: TCP Selective Acknowledgment Options. RFC 2018 (October 1996)
11. Murata, T.: Petri Nets: Properties, Analysis and Applications. Proc. of the IEEE 77(4) (April 1989)
12. Ns-2 network simulator homepage, http://isi.edu/nsnam/ns/
13. Petri Net Kernel and Petri Net Cube homepage, http://www2.informatik.hu-berlin.de/top/pnk/
14. Ye, Q., MacGregor, M.H.: Combining Petri Nets and ns-2: A Hybrid Method for Analysis and Simulation. In: 4th Annual Conference on Communication Networks and Services Research (CNSR), Moncton, New Brunswick, Canada, May 24–25 (2006)
15. Rząsa, W., Bubak, M.: Application of Petri Nets to Evaluation of Grid Applications Efficiency. In: Proc. of CGW 2008, Kraków, pp. 261–269 (2009) ISBN 978-83-61433-00-2
16. Rząsa, W., Bubak, M., Baliś, B., Szepieniec, T.: Overhead Verification for Cryptographically Secured Transmission in the Grid. Computing and Informatics 26, 89–101 (2007)
17. Transmission Control Protocol, RFC 793, Information Sciences Institute University of Southern California (September 1981)
18. Wei, D.X., Cao, P.: NS-2 TCP-Linux: an NS-2 TCP implementation with congestion control algorithms from Linux. In: WNS2 2006: Proceeding from the 2006 workshop on ns-2: the IP network simulator, Pisa, Italy. ACM, New York (2006)

# Adaptive Streaming of Stereographic Video

Krzysztof Grochla and Arkadiusz Sochan*

Institute of Theoretical and Applied Informatics of Polish Academy of Sciences,
Bałtycka 5, 44-100 Gliwice, Poland
{kgrochla,arek}@iitis.pl

**Abstract.** We present a tool for online compression and streaming of stereoscopic video and images and a consideration on adaptation of the video stream to network conditions. The software allows to use different encoding schemes for video compression and streams the stereo frames using UDP protocol. We give measurements of the frame delays in the transmission for different codec configurations.

## 1 Introduction

The digital stereo video is a sequence of stereo image pairs. Each pair contains image for left and right eye to produce a three-dimensional view of the scene. The main benefit over traditional video is the representation of depth of the image and providing the user with a more realistic representation of the view. The presentation of the stereographic video requires dedicated equipment to display the pictures for left and right eye separately. Stereographic monitors like Sharp LL-151-3D are available or, alternatively, projectors and monitors with special glasses – e.g. 3D Barco Galaxy projector. The online stereo video transmission may provide a 3-dimensional perception of a remote scene – it creates much deeper representation of the remote scene. The stereo video transmission may be realized as two simultaneous mono video transmissions, but it creates synchronization problems and the coding scheme will not utilize the similarity of frames to improve the encoding efficiency. Thus the stereo transmission generates slightly different traffic than the classical video streaming.

The rapid growth of the Internet over the last few years and the increase of available bandwidth has made the on-line transmission of video possible. Many web sites with video content have been created and the video streaming is becoming a popular service. The transmission of stereographic images however, has not become a common issue right now. The transmission of a video signal through a computer network is a complicated task. Subjective video quality depends on many conditions, some of them independent from the network (i.e. frame lossy compression ratio), while others highly dependent (i.e. inter-frame delay, accounting for smoothness of video presentation). To optimize the latter case, two important factors must be considered. Firstly, the popular IP network is known for high variability of the transmission channel and it also lacks

* This work was in part supported by Polish Ministry of Science and Higher Education grant no. 3 T11C 045 30.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 87–94, 2009.

the mechanisms for allocating network resources for some connection or class of traffic. Secondly, current video coding standards (i.e. MPEG-2) produce packet streams with characteristics that make traffic engineering difficult. There are different approaches to solving this problem. Firstly, there is active research on upgrading the Internet with quality of service (QoS) mechanisms [1,2]. Various traffic shaping algorithms are devised to improve the parameters of video transmissions [3,4,5]. New video compression techniques are tested, which add scalability to the stream, adding new possibilities for transmission control [6,7].

This paper is organized as follows. Section 2 contains information about encoding scheme and data stream creation. Section 3 contains the results of the test and the measurements of the frame delay. The last section provides a short summary of the article.

## 2 A Tool for Stereo Video Streaming

### 2.1 Coding/Decoding System

Coded video stream is a sequence of compressed video frames. Most of the compression methods use compressed stream of coefficients obtained by some image transform, the most popular being either the Discrete Cosine Transform (DCT) or Discrete Wavelet Transform (DWT). The traditional form of coding results in "closed" stream, which gives one option only – to decode completely. By splitting the coefficient data into groups, the number of options can be increased. Decoding all of the groups results in the exact reconstruction of the frame, while decoding only some of them results in the predictable degradation of quality. The latter value can be exactly estimated [8].

Coefficient data separation takes various forms. For DCT coefficients, the set of values for $8 \times 8$ block is either divided into groups ordered by increasing frequency, or by base level of reconstruction quality and successive corrections decreasing error level. DWT coefficients are normally coded one bit plane at a time, starting from the most important bits. The coding process either utilizes hierarchical statistical dependencies between coefficients (SPIHT) or outputs the given number of refinement streams for a block (EBCOT).

Depending on the way the refinement information is organized, there are three basic types of scalability: spatial (additional information increases the size of the image), temporal (additional information inserts more frames) and SNR (signal-to-noise; information improves the quality of the image). They can be combined to form complex profiles, spatial-SNR for example.

Popular video coding schemes MPEG (2 and 4) use DCT as a transformation. Data is organized into two layers – primary and extended. For MPEG-2, one can either decode primary layer, or both. For MPEG-4, the option is to decode the primary layer and any part (with single-bit resolution) of the extended layer. The latter scheme is called Fine Granular Scalability (FGS). Different combinations of profiles and levels allow broad range of scalability settings.

DWT as a base transform for image coding has been recently introduced in form of JPEG 2000 standard [8]. Although the algorithms are more complex than those of DCT, substantial improvement of coding quality can be attained. There is a body of research of DWT application to coding of video signals coding [6,7,9]. The results show that emerging technology will be superior to traditional MPEG in terms of coding efficiency and scalability possibilities.

Various types of 3D capturing and displaying techniques have been developed in order to produce the same depth sensation in virtual worlds as we experience in the real world. Many innovative studies on 3D images were focused on developing efficient compression algorithms including video streams encoding. MPEG-2 and MPEG-4 standards only partially support the encoding of stereoscopic images. In the MPEG-2 we can use Multiview profile, whereas in the MPEG-4 we can apply auxiliary frames. We propose our own method of encoding of stereoscopic pictures to 3D video streams that is more flexible and scalable than the solutions described in the standards. We applied two approaches for encoding of prediction frames: a channel (disparity) compensation and a temporal (motion) compensation. We implemented four types of video frames:

1. Intraframe Separated (IS) – a frame is encoded without prediction.
2. Intraframe Joined (IJ) – a frame is encoded applying the disparity compensation.
3. Prediction Separated (PS) – a frame is encoded applying the motion compensation.
4. Prediction Joined (PJ) – a frame is encoded applying both the disparity and the motion compensation.

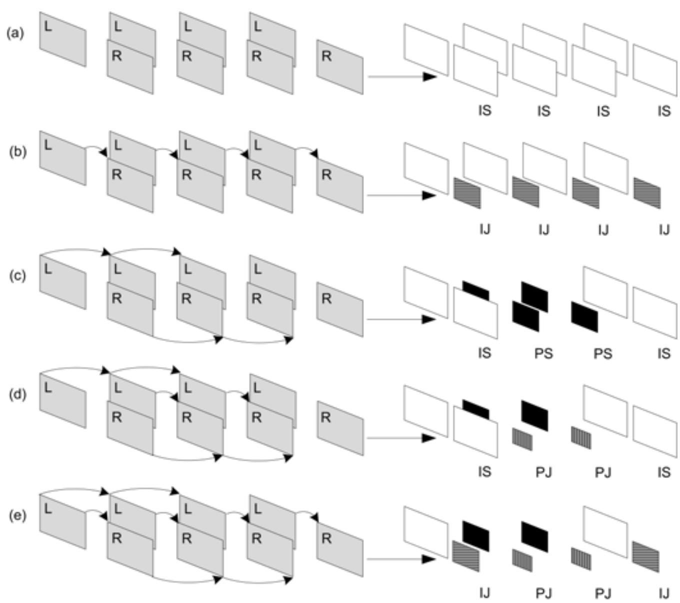



**Fig. 1.** Compression methods

We used the above mentioned types of frames to propose five schemes of stereoscopic videos encoding:

(a) a stream is encoded using IS frames only,
(b) a stream is encoded using IJ frames only,
(c) a stream is encoded using IS and PS frames,
(d) a stream is encoded using IS and PJ frames,
(e) a stream is encoded using IJ and PJ frames.

The schemes are illustrated in Fig. 1 We implemented both Discrete Cosine Transform and Discrete Wavelet Transform to encode stereoscopic images. However, we used only DCT-based encoding methods for experiments described in this paper.

### 2.2 Transmission of the Frames

To evaluate the different compression methods we have created custom stereo video streaming software. It uses UDP as a transport stream. After encoding the frames are divided into packets of the size corresponding to the MTU of the network. The data unit size can be manually adjusted. The frame header with information about the encoding method used, frame resolution and size is transferred by the first packet and a small header with frame number, sequence number and CRC is added to each of the packets. Then the packets are transferred to the client by any kind of network supporting IP protocol – it can be wired or wireless network. The client assembles the frame, decodes it and displays to the user. If some of the packets within the frame are missing, it can not

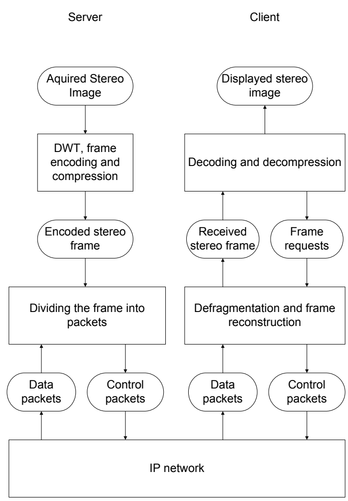



**Fig. 2.** Flow chart of the algorithm

be decoded – in this situation the frame is considered lost and the user sees the previous frame for slightly longer time.

The software was implemented under the Microsoft Windows operating system in .NET technology. It offers video streaming and basic video on demand functionality, like start, stop and pause of the transmission. There are two simultaneous communication channels – the data transmission from a server to a client and control connection from a client to a server. Every frame transfer is initiated by the client. The client transmits a control packet containing request for every frame. It allows to regulate and scale the stream to the observed network conditions. The client sends requests for each of the frames. The server sends data packets only in response to the request. If some of the frames are not received within the required time the client may limit the rate of request, to limit the bandwidth used by the transmission. The architecture of the software is presented in the Fig. 2.

### 2.3 Generation and Regulation of the Traffic

Most of the teletraffic networks today treat all kinds of traffic in the same manner. For a given connection, there are no guarantees on the value of the bit rate, or the time of transfer. This results in difficulties and a decrease in subjective quality of novel multimedia services (video on demand, teleconferences), where transmission time parameters are more important than rate of errors, for example.

Traffic shaping algorithms are used to control the load of the network (and with that to provide general QoS for the whole network). Their function is to delay or drop incoming transmissions of packets, depending on some estimated parameters (predicted network load for example). When properly applied, they reduce the number of lost packets, transmission delays or improve other QoS factors. An example of this kind of algorithm is Token Bucket Filter (TBF) used to limit and reduce the burstiness of data transmission. Other examples are RED, and SFQ algorithms [10].

Different kinds of transmission require different kinds of traffic shaping algorithms. For example, the video data stream quality is highly dependant on variations in packet arrival delays (jitter). An algorithm that introduces buffering and delays can decrease the QoS to unacceptable level, even though enough bandwidth is available. Normal data transmission, on the other hand, is sensitive to packet dropping techniques. With current compression techniques one can easily generate a stream that meets the target bit rate. Reducing of the bit rate even by as little as 5% however, produces significant jitter and quality degradation. The requirement of preserving constant bandwidth in the best-effort network environment is too strict, therefore some other solution must be adopted.

The partial answer is to use scalable video streams. Scalability in general is the possibility to increase or decrease the system performance as a result of changes in the environmental conditions. Adopting the definition to stereo streaming system with a traffic shaping mechanism that pushes the video data to the network, we can say that a scalable video transmission occurs when transmitted stream parameters (the size of the frames) can be altered to match the available network resources (measured in transmission time statistics for example). With a scalable video stream, the size of each frame can be adjusted by throwing away any part of it. It is a job of a dedicated traffic shaping algorithm to balance the reduced image quality (coming from smaller frames) and jitter (resulting from pushing large frames).

For efficient realization of the above approach, two conditions must be met. Firstly, the transmitted stream must be prepared in scalable form, discussed below. Secondly, there's a need for a traffic shaper that is able not only to discard but also to reduce the packet size. This is not possible with most of the current algorithms, which rely on the assumption of protecting packet's integrity. To support the scalability of the stream we added the regulation to the stereo video transmission. The algorithm is organized as follows: clients send requests for each frame on regular manner. The request contains the maximum value of the frame size in bytes, the value being constantly updated based on the frame arrival times. The inter-arrival times are stored as a description of network load. Generally, with an increase of the frame transmission times the frame sizes should be decreased. This means that also the quality parameters must be decreased. This relation is highly dependent on the network (number of transmissions, switches buffers' states), so it cannot be easily described. The requested frame size is determined by the relation between the previously requested sizes and their arrival times. The lower bound on frame size is externally defined as the minimum size when image is no longer readable; the upper bound is lossless coding frame size on the server side.

The network state is estimated on the basis of the delay between successive image packets. The server scales the video frame according to the client's request and pushes it into the network. The client measures the delays and calculates the frame request on the basis of this measurement. The algorithm has to take into account the delay between sending of the control information and its taking effect. In the current version of the software the fine scalability of frame sizes is not supported. The client regulates only which frames should be transferred – if the delay is too high to transfer all the frames the client does not send some of the requests.

## 3 Measurement Results

To evaluate the behavior of scalable stereo the software for stereo streaming was tested in real network conditions. We considered a typical case of home user connected by an ADSL link to the Internet and a stereo streaming server within IITiS PAN network. The packet round trip times between hosts were from 8 ms to 10 ms. The datarate of the ADSL link was 1 MBit/s. We used a sequence of 300 frames of stereo movie and measured the time between sending the request for the frame and the time when the complete frame was assembled. The results were collected for 5 different encoding schemes:

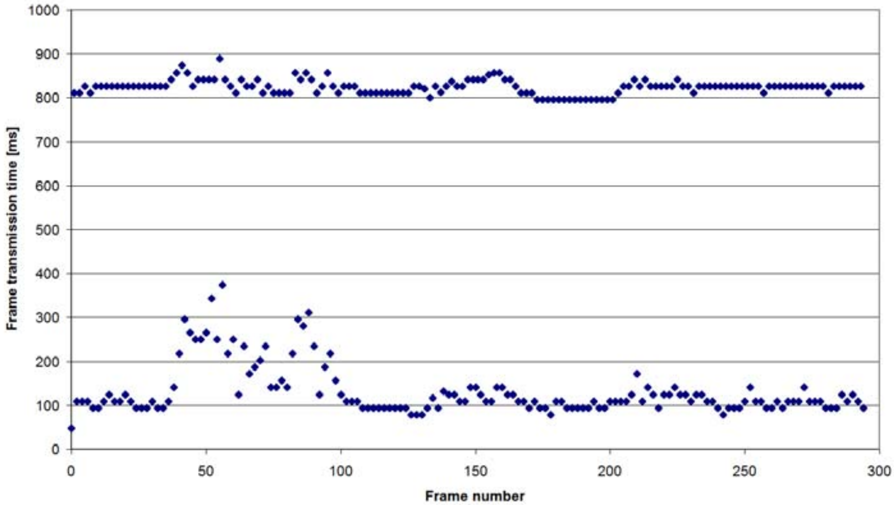



**Fig. 3.** Frame transmission times for IS_PS1 encoding

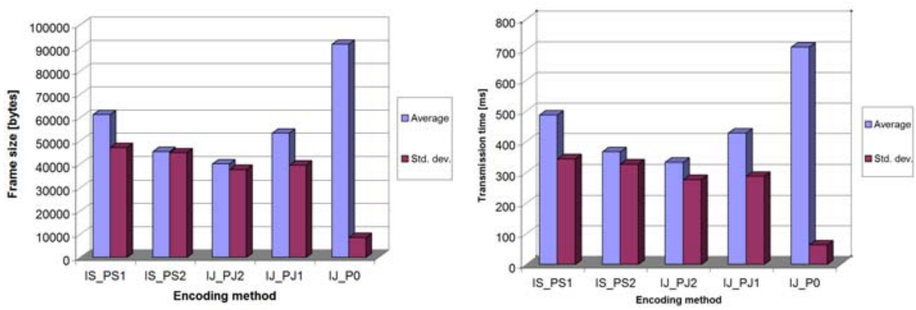



**Fig. 4.** Average frame sizes (on the left) and transmission times (on the right) for different encoding algorithms

**IS_PS1** – one PS frame between IS frames,
**IS_PS2** – two PS frames between I frames,
**IJ_PJ1** – one PJ frame between IJ frames,
**IJ_PJ2** – two PJ frames between IJ frames,
**IJ_P0** – only IJ frames.

The plot of frame delays per IS_PS1 encoding scheme is shown in Fig. 3. The codec generates one big frame and one small frame alternately, thus creating a significant difference in frame transmission times. The plots in Fig. 4 present the comparison of frame sizes and frame delays for different encoding schemes. The average values for the whole transmission were calculated together with standard deviation. It can be observed that the best results were achieved for IJ_PJ2 coding scheme. Unfortunately, the variation of frame transmission times is very high for this method. The most constant delay was generated by IJ_P0 scheme, but the drawback is that the delay was the highest of all tested methods. The average frame transmission time is linearly proportional to the frame size, but the standard deviation for transmission is higher, as the network may generate additional delays.

## 4 Summary and Future Work

We present the software for encoding and transmission of stereo video streams. The average frame delays have been evaluated in a real network environment and the frame transmission delays for different encoding schemes have been compared. As future work we want to introduce the scalability support for automatic adjustment of the datarate to existing network conditions and include forward error correction codes into the transmission.

## References

1. Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W.: An architecture for differentiated services rfc 2475. Technical report, The Internet Society (1998), http://www.faqs.org/rfcs/rfc2475.html
2. Kihong, P., Willinger, W. (eds.): Self Similar Network Traffic and Performance Evaluation. Wiley and Sons, Chichester (2000)
3. Chaddha, N., Wall, G., Schmidt, B.: An end to end software only scalable video delivery system. In: Little, T.D.C., Gusella, R. (eds.) NOSSDAV 1995. LNCS, vol. 1018. Springer, Heidelberg (1995)
4. Park, S.H., Ko, S.J.: Evaluation of token bucket parameters for vbr mpeg video transmission over the internet. IEICE Transactions on Communications E85-B(1) (2002)
5. Atiquzzaman, M., Hassan, M.: Adaptive real-time multimedia transmission over packet switching networks. Real-Time Imaging 7(3) (2001)
6. Bottreau, V., Benetiere, M., Felts, B., Pesquet-Popescu, B.: A fully scalable 3d subband video codec. In: Proc. IEEE Conference on Image Processing 2001 (ICIP 2001) (2001)
7. Luo, L., Li, J., Li, S., Zhuang, Z., Zhang, Y.Q.: Motion compensated lifting wavelet transform and its applications in video coding. In: Proc IEEE Int. Conf. on Multimedia and Expo (2001)
8. ISO/IEC: 15444-1: Jpeg 2000 image coding system. Technical report, International Organization for Standarization (2001)
9. Choi, S.J., Woods, J.: Motion-compensated 3-d subband coding of video. IEEE Trans. Image Proc. 8(2) (1998)
10. Grochla, K., Głomb, P.: Kształtowanie ruchu w routerach pracujacych w systemie Linux. Studia Informatica 23(2A) (2002)

# The Influence of Electromagnetic Disturbances on Data Transmission in USB Standard

Michał Maćkowski

Silesian University of Technology, Institute of Computer Science,
Akademicka 16, 44-100 Gliwice, Poland
michal.mackowski@polsl.pl
http://www.polsl.pl/

**Abstract.** Universal Serial Bus (USB) architecture is becoming a popular substitute for parallel or serial bus RS-232 architecture. In addition to parallel and serial ports, USB ports are provided as standard equipment on most computers manufactured today. There are also more and more network devices equipped with USB interface, for example USB Ethernet adapter, network Wi-Fi USB card, or USB adapter to connect two computers. Although the published USB standards require all devices with the USB logo meet today's PN-EN requirements for "CE" mark, very few commercially available USB peripherals actually do so, on the immunity side. This research focuses on the influence of electromagnetic disturbances on data transmission in USB standard in accordance with electrical fast transient immunity test and can be easily expanded to other communication standards.

## 1 Introduction

The demand for capacity of communication interface increases together with the development of computer systems. A wide variety of peripheral devices impose diverse communication requirements, and standards existing for many years, such as RS-232 or IEEE-1284, could no longer satisfy the market.

The generality and growing popularity of USB standard caused the market to become full of network devices connected to a computer via the USB interface. The examples of such devices are: USB Ethernet adapter, network Wi-Fi USB card, USB adapter to connect two computers. While using the USB interface, as the element that connects for example Fast Ethernet network and a PC computer, it is necessary to answer the question whether the interface is susceptible to electromagnetic disturbances, therefore whether using it would not cause the corruption of transferred data.

This is the reason why conducted research focuses on interference of the data transmission in USB standard, caused by the influence of electromagnetic field. However, they can be easily generalized on the other communication interfaces. The main target of the study is to verify the immunity of the USB standard for electromagnetic disturbances, moreover the study tries to compare USB immunity among others communication interfaces. In contrast, the study [2] presents the test results of the Fast Ethernet immunity for electromagnetic disturbances.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 95–102, 2009.

Our research aim at improving the methods of protecting devices in case of incorrect operation. This may be caused by the electromagnetic disturbances that interfere through: power systems, ground and interface circuit (transmission lines). The research already being conducted will allow to determine the immunity of already existing systems and computer devices. Moreover it will enable to determine the compatibility of those devices with the established standards harmonized with EMC (electromagnetic compatibility) directive. The very important feature of the research itself is the opportunity to check the influence of electromagnetic interference on data transmission in communication standards.

## 2 USB Standard

The development of computer systems as well as still increasing group of electronic devices being able to exchange the information with a computer, made the vendors elaborate the communication standard which would ensure an easy way of connecting devices to the system, and at the same time would be very functional and multi-purpose.

Companies like Microsoft, Intel, Compaq, IBM, DEC released USB, which was the solution to the hitherto problems connected with the limited amount of communication ports: RS-232, LPT IEEE 1284 and with the limited data transmission speed of those ports.

The kind of transfer used for exchange of the information between the computer (host) and the executive device is the basic element which determines the class of devices equipped with USB interface. Because of the fact that there is a huge variety of devices realizing different communication requirements, four types of transfers were distinguished: control, interrupt, bulk and isochronous.

In the conducted research for data transmission, bulk transfers have been used because they guarantee the integrity of data (error checking) and the opportunity of getting the maximal speed of USB. However, this is possible only under the condition that no other kind of transfer would use USB in the same time. Another

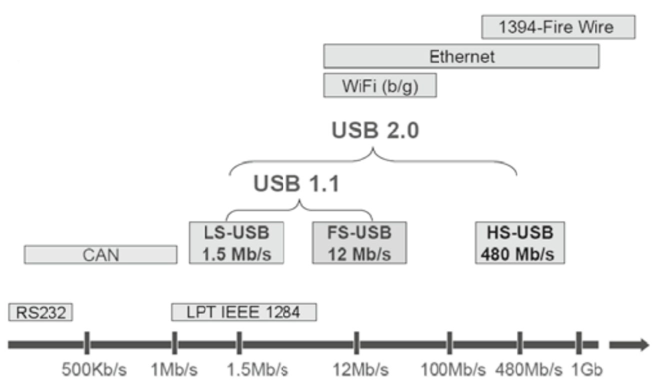



**Fig. 1.** Comparison of the most available communication standards

kind of transfers used in the research were control transfers, being the obligatory element of each device equipped with USB communication interface. Control transfers are also used in control process.

Figure 1 presents the comparison of available communication standards. It can be noticed that thanks to differential speed of transmission fitted to different kind of devices, USB standard complies largely with the requirements of users as well as vendors of computer equipment, offering the rate of data transmission from 1.5 Mb/s up to 480 Mb/s.

## 3 Test Bench and Research Procedure

In order to prepare the test bench, it was necessary to create the device equipped with USB interface, through which a device communicates with a computer. Construction of the own device ensures a full control over transmitted data, and what is more important the opportunity to modify device software and to adjust it to our needs.

The test bench consists of equipment and software created on the device and computer sides. Figure 2 shows model of the communication layer implemented on the computer and the device sides.

Following blocks: USB Host Hardware and PIC18F4550 physical interface USB correspond to the lowest, physical layer with regard to ISO model. Meanwhile the rest of the layers correspond to the software on the PC computer and USB interface device side. The device designed and constructed for the project was based on 8-bits Microchip microcontroller PIC18LF4550. This microprocessor contains integrated USB controller compatible with 1.1 standard version and works with Low Speed 1.5 Mb/s and Full Speed 12 Mb/s. Depending on the device software (firmware) it is being recognized by the computer operating system as the HID (Human Interface Device) or MSD (Mass Storage Device) class device.

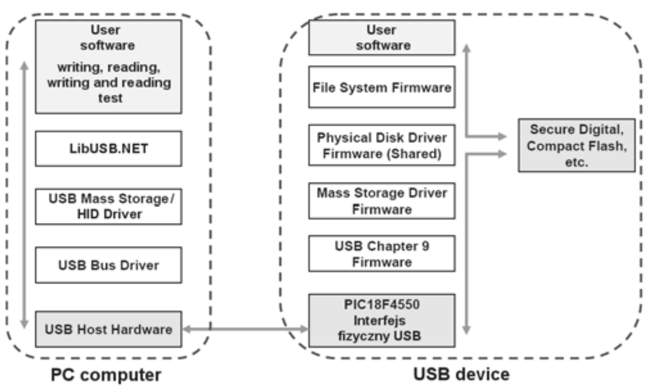



**Fig. 2.** Communication layer model implemented on the computer and device sides

The software part of the project consists of two applications. The first is the operating system of the device and the second is the application on the PC computer side. The operating system implemented on the device uses the mechanism of interrupts to serve USB controller. The program and data memory store all of the structure defining device parameters. These parameters determine the class of the device.

During the research it was also necessary to write a PC program which would allow to measure rate of data transmission. The program was created in C# language in Microsoft .NET and used Open Source LibUSB.NET library. The other features of the program are: test selection (writing, reading, writing and reading test) and displaying the information about device, configuration, interface, endpoint descriptors.

### 3.1 Test Bench

There are four items that can be distinguished in test position: two PC computers, USB device and USB protocol analyzer Explorer 200 by Ellisys (Fig. 3). The role of analyzer is to sniff all frames transmitted between PC and the device. Consequently these frames are transferred to another computer-analyzer, which contains software for analyzing sniffed frames.

A USB wire is inserted into capacitive coupling clamp that is used to couple EFT (Electrical Fast Transients) bursts onto I/O lines. According to PN-EN 61000-4-4 standard, during the test, impulsive disturbances are injected simultaneously to transmitter and receiver.

The proper placing the clamp which injects disturbances allow to preview corrupted frames of data packets in Ellisys Visual USB application installed on the computer-analyzer. During the analysis the USB wires of 3 m and 5 m length were used to transmit the data in Full Speed mode. The two tests were conducted:

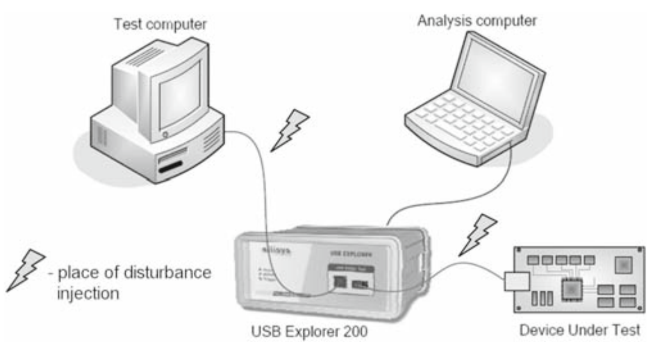



**Fig. 3.** Test position for analysing transmitted data using the USB protocol analyzer Explorer 200 by Ellisys

– Data write test (OUT operation). Direction of the transmission from the computer to the USB device. Clamp was placed between computer and USB analyzer.
– Data read test (IN operation). Direction of the transmission from the USB device to the computer. Clamp was placed between USB analyzer and the device.

### 3.2 The Research Procedure

One of the targets of this research was to test the data transmission immunity in USB Full Speed standard for the electrical fast transient bursts. The test was performed in accordance with standard PN-EN 61000-4-4 that is harmonized with EMC directive 2004/108/WE. Series of electrical fast transient disturbances are designed to simulate a relay contact bounce in the electric network.

Series of disturbing impulses are induced in transmission lines through a clamp. The research parameters being compatible with the standard are as follows:

– repetition frequency 5 kHz,
– burst duration 15 ms,
– burst period 300 ms,
– output voltage range change from 250 to 1000 V – according to level 1 to 3 in PN-EN 61000-4-4 standard.

Level one refers to requirements for devices working in residential, commercial and lightly industrialized environment, whereas levels two, three and four refer to requirements for devices working in the industrial environment. Table 1 presents levels of the sharpness test according to the standard.

**Table 1.** Test levels for the Electrical Fast Transient test, according to PN-EN 61000-4-4

| Level | Power ports | | I/O ports | |
|---|---|---|---|---|
| | Voltage peak [kV] | Repetition rate [kHz] | Voltage peak [kV] | Repetition rate [kHz] |
| 1 | 0.5 | 5 | 0.25 | 5 |
| 2 | 1 | 5 | 0.5 | 5 |
| 3 | 2 | 5 | 1 | 5 |
| 4 | 4 | 2.5 | 2 | 5 |
| x | Special | Special | Special | Special |

## 4 The Influence of Electromagnetic Disturbances on Data Transmission in USB Standard

The research was initiated for disturbing signals whose peak value equals 250 V, which is compatible with level one according to PN-EN 61000-4-4 standard. For such test parameters USB standard is characterized by almost complete immunity for fast electrical transients. The only thing that was noticed during the research was some destroyed transactions, being the part of frames. Retransmission of those tractions had low influence on data transmission speed. The USB standard was equipped with a range of mechanisms which are designed to protect it from incorrect work of the system. The main tasks of elements that protect USB are controlling the data correctness, limiting the time for response and the mechanism of data toggling (USB uses this mechanism as a part of its error correction).

The main way to ensure the correct work of the system is to detect the errors in received packet and not to answer (ACK) in case of their presence. When the sender does not receive a confirmation, then the damaged frame is retransmitted. This mechanism is very effective, especially in the case of distortion in data transaction area. However, if a mistake appears in the response sent off to a receiver, then a problem arises and the best solution is to use data toggles. It is the mechanism of alternating numeration of packets, which protects USB standard from losing the synchronization.

The next step of the research was to test immunity of USB standard for levels 2 and 3, which correlates with successive amplitudes of disturbing signal on level 0.5 kV and 1 kV. Figure 4 shows the fragment of a frame where transmission error appeared during writing operation into the device. The figure presents also the content of data packet field (Data 1), where next 57 bytes of data were corrupted.

The majority of corrupted transactions were the effect of distortion of content of data packet. Each transaction in USB system consists of token packets, which determine direction of data transmission (writing/reading), data and acknowledge packets. During the test the maximum acceptable packet size was used, which equals for Full Speed 64 bytes. A full-size of data packet is related to fewer amounts of tokens and acknowledges packets appearing in USB.

Nevertheless, the distortion of single bit in data area requires the retransmission of all 64 bytes data block. It was observed that in all available devices the maximum size data packet is used in order to get the maximum speed of transmission. Thus, during the test such type of data packet was used.

Figure 5 presents the content of frame in USB standard for reading operation (direction of transmission is from the device to the computer). It can be seen that USB standard is no longer immune when the amplitude of disturbances is 500 V or higher. There are no charts presenting decreases of transmission speed during writing and reading operation. It is due to the shortage of immunity of standard for the value of disturbing signal voltage 500 V or higher.

It was also observed that the device is completely disconnected from a system when the amplitude of disturbances is 0.5 kV or higher, and the previously presented mechanisms became useless. The explanation why this situation occurs may be traced to overlapping of the frequency of appearing consecutive initial parts of frames SOF (Start Of Frame) in USB standard with the frequency of disturbing signal determined by standard PN-EN 61000-4-4. The frequency of

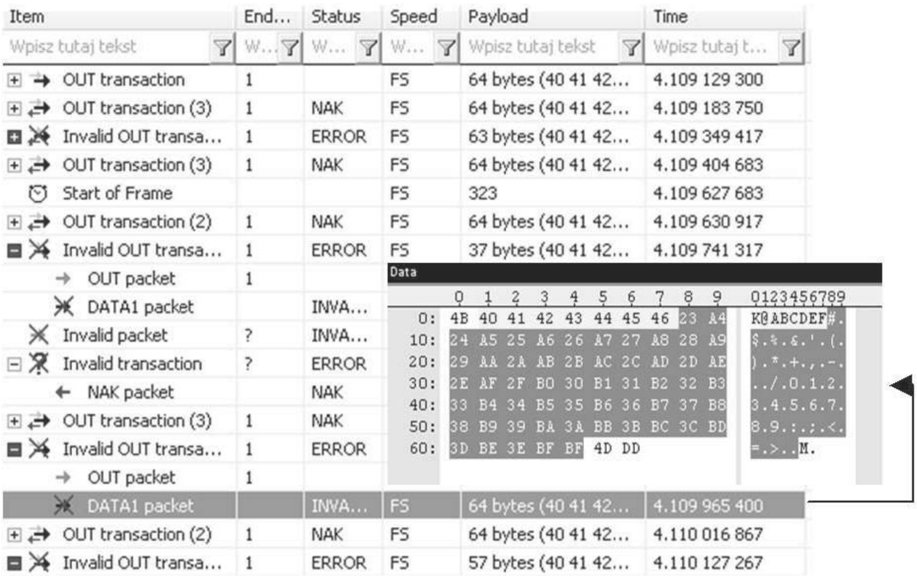



**Fig. 4.** The results of frame analysis during the writing operation. Test conducted for level 1 – peak voltage 500 V.

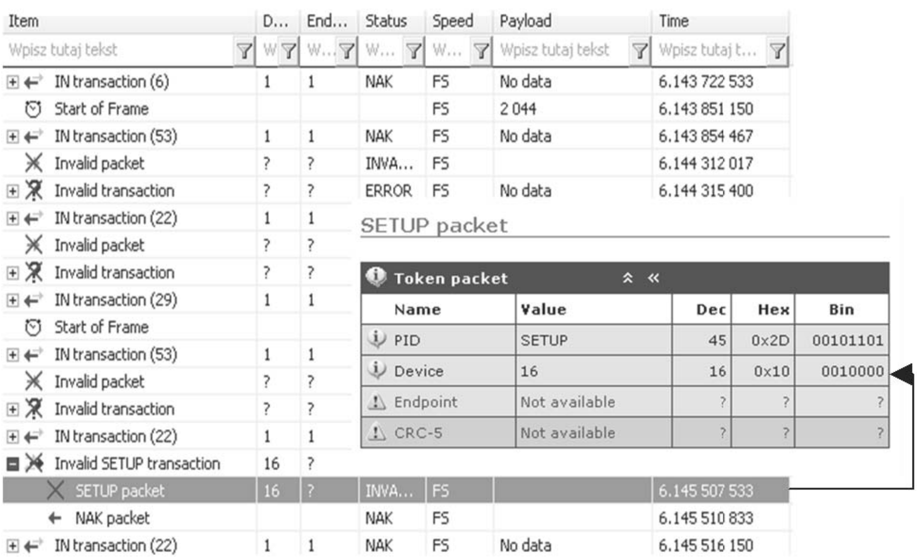



**Fig. 5.** The results of frame analysis during the reading operation. Test conducted for level 1 – peak voltage 500 V.

appearing disturbing impulses in each 15 ms interval equals to 5 kHz, which in practice causes the loss of following 15 frames in USB protocol, and finally the loss of the connection with the computer.

## 5 Conclusions

The study focuses mainly on interference of data transmission in USB standard caused by electrical fast transient disturbances according to PN-EN 61000-4-4 standard. The main purpose of the study was to test the immunity of USB standard and its mechanisms for electromagnetic disturbances.

Despite using screening wires for data transmission and symmetric transmission while designing the device with USB interface, there is a problem with connection of the screen on the device side. Most vendors connect screen directly to circuit ground on printed board. Thus, the disturbances inducted on a screen interfere with electronic circuits which deteriorate the work of devices. Additionally, USB not only transmits data but also provides power supply. Thus, the disturbances can propagate simultaneously to electronic devices through data and power lines and cause interferences in transmission channel.

The conducted research revealed the high susceptibility of USB standard for electromagnetic disturbances, especially when the amplitude of disturbances impulses is higher than 500 V. The result of such disturbances is that devices are no longer visible in a system and the complex procedure of attaching the device to a computer makes it impossible to recognize the device again in the system without human interaction.

As the test results show, the high susceptibility of USB standard to electromagnetic disturbances can cause additional interferences of data transmission. In publication [4] and [6] the issues referring to the immunity of other communication standards such as: RS-232 and Fast Ethernet were presented.

## References

1. Anderson, D.: Universal Serial Bus System Architecture. Mindshare, Inc., Addison-Wesley Developers Press (1997)
2. Flak, J., Skoroniak, K.: Badania odporności standardu Fast Ethernet na zaburzenia przewodzone. In: XV Krajowa Konferencja Sieci Komputerowe. Zakopane (2008)
3. Hyde, J.: USB Design by Example, A Practical Guide to Building I/O Devices. Intel Press (2001)
4. Maćkowski, M., Ober, J., Skoroniak, K.: Fast Ethernet a zakłócenia elektromagnetyczne. In: XIV Krajowa Konferencja Sieci Komputerowe. Zakopane (2007)
5. Mielczarek, W.: USB Uniwersalny interfejs szeregowy. Helion, Gliwice (2005)
6. Ober, J., Skoroniak, K.: Wpływ zaburzeń elektromagnetycznych na transmisję danych w standardzie RS232C. In: XIII Krajowa Konferencja SCR. Ustroń (2006)

# Electromagnetic Emission Measurement of Microprocessor Units

Michał Maćkowski and Krzysztof Skoroniak

Silesian University of Technology, Institute of Computer Science,
Akademicka 16, 44-100 Gliwice, Poland
{michal.mackowski,krzysztof.skoroniak}@polsl.pl
http://www.polsl.pl/

**Abstract.** Each microcontroller realizing the code saved in a program memory emits electromagnetic disturbances both conducted, that propagate through lines connected to the processor, and radiated in the form of electromagnetic field. All of these undesirable signals emitted by the processor can be measured, received and interpreted by the appropriate methods.

This paper illustrates the research results concerning the character of signals emitted by the selected microcontroller via the power supply lines. The purpose of the study is to determine the spectrum of signal emitted by the processor depending on the instruction being realized. The research results presented in the study indicate that there are the differences in the spectrum of the signal emitted by the processor, depending on the program being executed. These results are the subject for further research.

## 1 Introduction

Fast technological progress, development of computer systems, increasing the data transmission speed and high reduction of electronic system size, requires putting the emphasis on research connected with immunity and electromagnetic disturbances emission.

The constant need for more and more miniaturization and integration, led to very large scale integration circuits. The high integration scale and permanent increase of the frequency of microprocessor circuit, entails that current peaks are generated with higher amplitudes and shorter rise times on the power supply and I/O lines of the electronic circuits. These impulses are generated by simultaneous switching of millions of transistors inside of the integrated unit. Propagation of such currents via wires and paths on a PCB (Printed Circuit Board) into other electronic devices may cause the problems with their proper work.

Microprocessor units, presently used as controllers in network devices, can be also considered as advanced chips being responsible for data processing and reconstruction of transmitted frames. What is more, such units can be also a source of electromagnetic disturbances, which interfere with operation of other electronic devices. Model of microprocessor executing specific instructions can

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 103–110, 2009.

be used to predict the emission of electromagnetic disturbances generating by microcontroller. The issue referring to this problem was raised in publications [1] and [3].

The main purpose of this study is to analyze and compare the measured spectrum of disturbing signal generated by microcontroller, depending on a program that it currently realizes. The microcontroller and the equipment used in the test are presented in Sect. 3 – test bench and research procedure. All tests were based on PN-EN 55022 standard harmonized with EMC (electromagnetic compatibility) directive 2004/108/WE.

## 2 Electromagnetic Interference

The fundamental parameter of electromagnetic disturbance is its frequency. Basically, EMC standards cover the range from 0 Hz to 400 GHz. However, not all frequency ranges are completely regulated by standards.

The first important range (Table 1) is a range about 50 Hz related to power network frequency. It is the range of harmonic current measurement, being the multiple of power network frequency. From the final part of this range to 9 kHz there is a frequency range that is not regulated by any standards. Above 9 kHz begins the range of high frequency, which is the main interest of this publication. This range is called radio frequency (RF).

The range of radio frequency is generally divided into two sub ranges: conducted and radiated. When the RF disturbance is lower, then the disturbance is considered as conducted, but on the other hand, when it is higher the disturbance is radiated. The conducted RF range between 150 kHz and 30 MHz is regulated by standards. The top frequency 30 MHz is then the beginning of radiating range. The upper limit of the RF radiation band depends on a standard and a device but usually it is a range to 1 GHz.

Electromagnetic interference (EMI) is caused by undesirable radiated electromagnetic fields or conducted voltages and currents. The interference is produced by a source emitter and is detected by a susceptible victim via a coupling path. Depending on the way they are reaching the device they can be divided into:

- Conduction emission – electric current,
- Radiation emission – electromagnetic field.

**Table 1.** Frequency range considered in terms of electromagnetic compatibility

| | **Frequency Range** | **Standard regulation** |
|---|---|---|
| **Harmonics** | 50 Hz – 2/2.5 kHz | regulated |
| **LF (Low Frequency)** | 2/2.5 – 9 kHz | not regulated |
| **Conducted RF (Radio Frequency)** | 9 kHz – 150 kHz | regulated for some products |
| | 150 kHz – 30 MHz | regulated |
| **Radiated RF (Radio Frequency)** | 30 MHz – 1/2/3 GHz | regulated |
| | more than 3 GHz | regulated for some products |

Conducted noise is coupled between components through interconnecting wires such as through power supply and ground wires or PCB tracks. Common impedance coupling is caused when currents flow from two or more circuits through the same impedance such as in power supply and ground wires.

## 3 Test Bench and Research Procedure

Test bench consists of a test device and equipment used for measuring conducted emission. The equipment under test (EUT) is 8-bit microcontroller PIC18LF4550 produced by Microchip. During the test the processor is working using 8 MHz internal oscillator and the operating voltage 5 V. According to microcontroller specification, 100 nF capacitor was added to the power supply lines. On the test board only power pins Vdd and Vss were exposed to simplify the test model.

In order to measure conducted emission, it was needed to insert Line Impedance Stabilization Network (LISN) between power supply network and test device. The main task of LISN is to lead out RFI signals from tested device to measuring input of EMI receiver (Fig. 1).

During the research the measuring receiver ESCI and Line Impedance Stabilization Network ENV216 produced by Rohde & Schwarz were used. Furthermore, the LISN performed additional tasks on the measuring channel. The most important tasks are as follows:

– Filtering the power network disturbances and not allowing them to propagate on high frequency output of LISN, that is to EMI input receiver,
– Filtering the emission disturbances generated by test devices and not allowing them to propagate to power network and pass them to high frequency output of LISN,
– Ensuring the impedance compatibility of measuring receiver and supply power circuit.

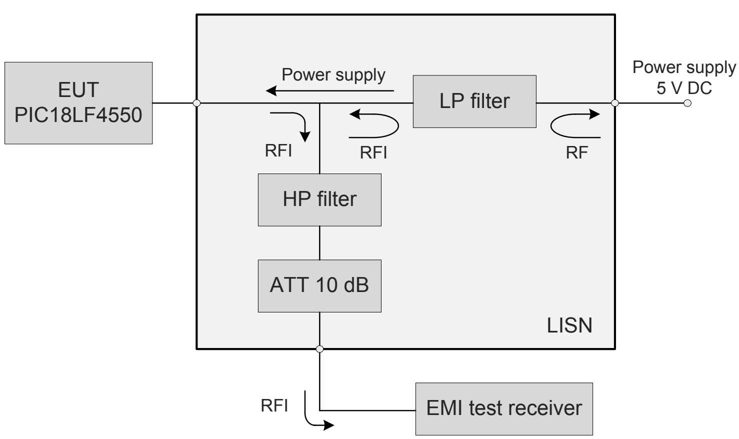



**Fig. 1.** The schema of research position

**Table 2.** Microcontroller test programs

| Program 1 | Program 2 | Program 3 | Program 4 | Program 5 |
|---|---|---|---|---|
| main | main | main | main | main |
| | NOP | ANDLW 0Fh | SUBLW 20 | MOVLW 85h |
| goto main | goto main | goto main | goto main | goto main |

As it was mentioned before, all of research was conducted in accordance with PN-EN 55022 standard. In this regulation, information referring to location of tested device in relation to reference ground plane (in this case it was a metal wall) and line impedance stabilizing network were included. It also contained information on the way of wire arrangement on the measuring table. On top of that, the standard describes the procedures concerning measuring of disturbing signals, emitted by devices and maximal acceptable level of these signals. The aim of this standard is to unify the requirements referring to disturbances levels, generated by devices within the scope of the standard. According to requirement the research was conducted for all measuring range from 150 kHz to 30 MHz.

The purpose of measurement was not to test whether the measured value of conducted emission exceeds acceptable levels determined by the standard. It allowed to simplify the measuring process and to reduce the measuring time. In order to measure the conducted emission depending on software realized by microprocessor, the detector of average value can be used. Such measurement will allow to determine the shape of conducted disturbances spectrum, emitted by device.

During the research the measurement of microcontroller conducted emission were carried out on its supply lines with utilization of five simple programs (Table 2). In practice, all software included in the program memory of processor, works in infinite loop, thus the test programs were constructed in the similar way. Program 1 is the reference point and consists of only one jump instruction – a loop. The remaining four programs run inside infinite loop and they realize some instruction that is different in each program.

The spectrum of disturbing signal emitted by the processor in program 1 was subtracted from the spectrum of disturbing signal determined for each of programs: 2, 3, 4 and 5. In this way charts of the difference of two spectra presenting the character of single instruction realized inside goto loop were received.

## 4 The Test Results

Charts presented in Fig. 2, 3, 4, and 5 show the difference of two spectra. In each case it is the difference of spectrum of signal emitted by the processor during the execution of program 2, 3, 4, 5, and spectrum of signal emitted by the processor when it realizes program 1 (Table 2). Program 1 is the point of reference to the rest of programs.

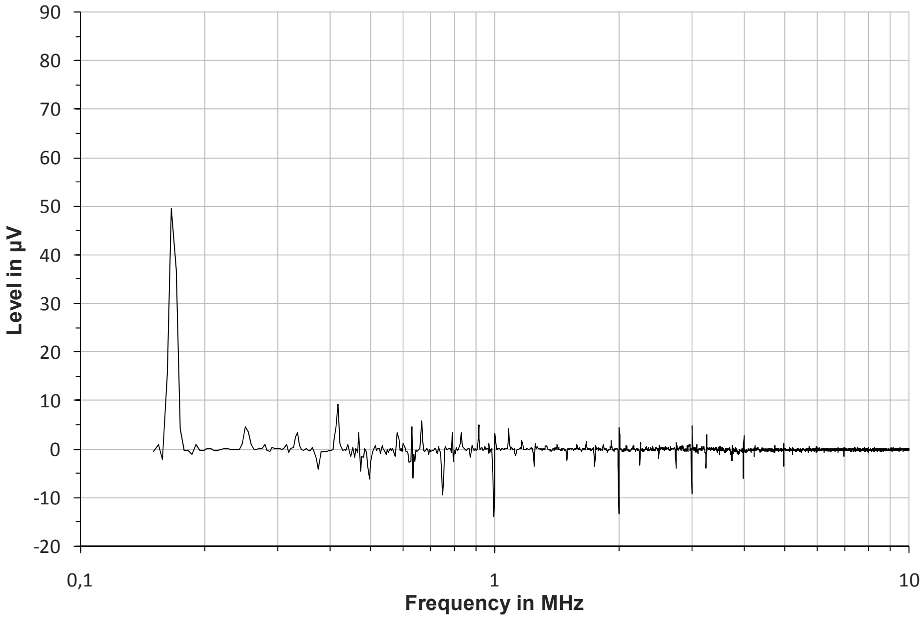



**Fig. 2.** The difference of spectra of signals for programs 2 and 1

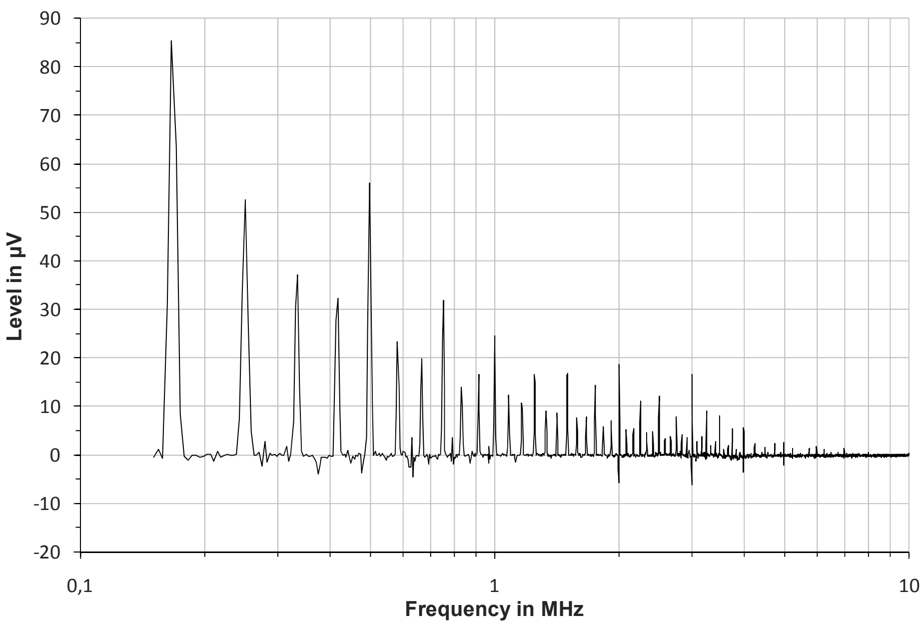



**Fig. 3.** The difference of spectra of signals for programs 3 and 1

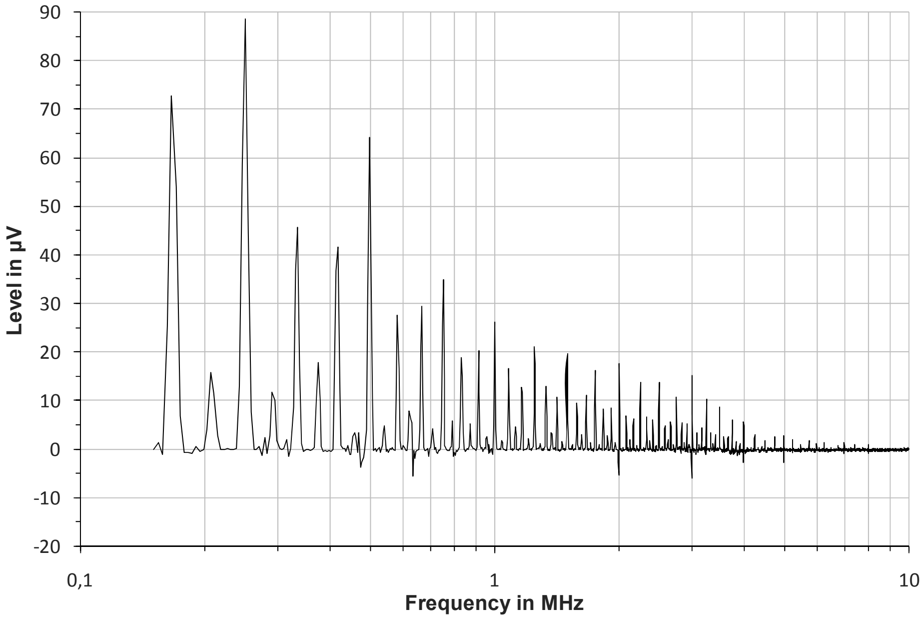



**Fig. 4.** The difference of spectra of signals for programs 4 and 1

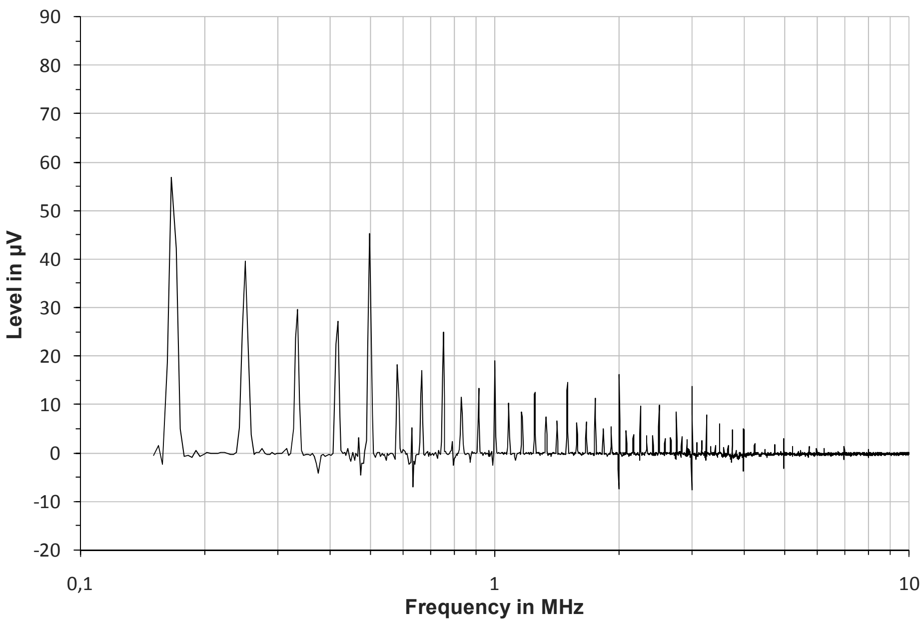



**Fig. 5.** The difference of spectra of signals for programs 5 and 1

The authors decided to present the difference of two spectra in each case, to eliminate all stripes from charts, which are permanent, and that appear in all of the research. These peaks appearing in spectrum, result from the frequency of the processor clock, and yet from other periodically realized operations. They have no influence on this study, thus they were eliminated from the research by the difference operation.

Since the processor works with the low frequency clock 8 MHz (time of instruction cycle 500 ns), therefore the charts present the spectrum difference of conducted disturbances emitted by device only in range from 150 kHz to 10 MHz. As the observations show, above this range the spectrum differences resulting from realization of different programs were insignificant. Hence, in all presented charts in this section the spectrum was limited to 10 MHz.

## 5 Conclusions

The presented results indicate clearly differences appearing in spectrum of conducted signals, emitted by the processor, that depend on the program. The research results discussed in the study refer to defined processor instructions. It can indicate then, that different instructions executed by the microprocessor cause that the emission of disturbing signals have different character and form. All instruction used in presented research were conducted during one instruction cycle. Due to this fact, they were chosen for tests.

All research conducted so far, intended to determine the spectrum of signal emitted by the processor, during realization of a particular single instruction. Nevertheless, this instruction was realized in the loop. Determining, on the other hand, a spectrum for signal emitted by the processor during execution of a single instruction that is not realized in the loop is the issue for further research of authors. This issue requires testing of the waveform of signals emitted by the processor in the time domain through power supply lines. Next, it is necessary to distinguish from this waveform only these parts of signal which are responsible for particular instructions, and to perform Fourier analysis for each of these parts. Based on these facts, it is possible to analyse the spectrum of signal emitted by the processor during realization of a single instruction. After the analysis, it will be probably possible to predict the instruction that is being presently executed, based on the signals emitted by the processor.

It is necessary to notice that the authors of publication [3] try to predict the emission of disturbing signals emitted by the processor on the power supply lines under different program behaviour. This issue is opposite to the one discussed in this study.

If the further research reveals that on the basis of signals emitted via power supply lines, there is a possibility to determine the code of a program realized by the particular processor, then the conclusions may indicate the existence of a serious danger to the programs which are save in the microcontroller memory. Moreover, in some case these programs should be protected from being copied.

## References

1. Bendhia, S., Labussiere-Dorgan, C., Sicard, E., Tao, J.: Modeling the electromagnetic emission of a microcontroller using a single model. IEEE transactions on Electromagnetic compatibility (2008)
2. Charoy, A.: Kompatybilność elektromagnetyczna: Zakłócenia w urządzeniach elektronicznych. WNT, Warszawa (1999-2000)
3. Chen, C.K., Liao, S.S., Sicard, E., Yang, C.F., Yuan, S.Y.: EMI prediction under different program behavior. In: IEEE EMC Symposium, Honolulu, USA (2007)
4. Hasse, L.: Zakłócenia w aparaturze elektronicznej. Radioelektronik, Warszawa (1995)
5. Machczyński, W.: Wprowadzenie do kompatybilności elektromagnetycznej. Wydawnictwo Politechniki Poznańskiej, Poznań (2004)

# Software Influence Upon AX.25 Protocol Performance

Bartłomiej Zieliński

Silesian University of Technology, Institute of Computer Science,
ul. Akademicka 16, 44-100 Gliwice, Poland
Bartlomiej.Zielinski@polsl.pl

**Abstract.** The paper presents results of effective throughput measurements in an experimental amateur Packet Radio network. We tested how software type and version used in transmission hardware, such as TNC controller, influences on network performance. We selected few software types and versions and tested them on two common hardware platforms based on Z80 microprocessor. In this way it was possible to find out if unsatisfactory efficiency of some TNC controllers network results from lack of their processing power or limitations caused by transmission software.

## 1 Introduction and Related Work

AX.25 protocol [1] belongs to the HDLC protocol family and is used as a data link layer in the amateur Packet Radio network that can be considered as an example of a simple wireless wide area network. Transmission hardware designed for Packet Radio, namely TNC (*Terminal Node Controller*), may also be used as examples of protocol converters that allow for integration of wired and wireless network segments [2]. It can be used, among others, in some telemetry or remote control networks, including those operating according to APRS (*Automatic Position Reporting System*) [3] protocol requirements.

Packet Radio network, as a solution of radio amateurs, has never been popular, which is acknowledged by small number of literature covering this subject. In particular, there is lack of in-depth analysis of factors that may influence on effective network parameters, such as throughput or transmission delays.

In previous works, we presented an analytical efficiency estimation of AX.25 protocol [4]. This analysis allows estimate how certain protocol parameters influence on its efficiency and – as a result – effective transmission speed observed by a user. We also developed an analytical model of TNC controller [5]. It allows estimate how presence of TNC controller influences on time parameters of the network (e.g., effective throughput or transmission delays). The model can also be helpful in buffer size selection.

The aforementioned works concentrate on theoretical analysis. They are useful when it is necessary to estimate, for example, maximum achievable throughput for a given parameter set; however, they do not take into account properties of transmission hardware and software. In fact, transmission hardware, such as

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 111–118, 2009.

TNC controllers [6], may be built using various types of microprocessors, running at various clock frequencies. They may also be equipped with a data memory of various capacity, thus buffer sizes may differ between individual TNC's. On the other hand, there are several software types and versions. During experimental tests in a Packet Radio network, we found out that despite relatively low transmission rates (few to few tens kbps), processing power of a microprocessor running in TNC controller has visible influence upon effective throughput. We also found out that there were some differences in protocol implementation details which were also significant from this point of view. Thus, we decided to check how effective throughput depends on a TNC software. There are several software types and versions available for Z80-based TNC's so it was possible to check and compare their behaviour on a common hardware platform.

## 2 Experimental Tests and Results

The experimental test were conducted using four types of Zilog Z80-based TNC controllers running at various clock frequencies $f_{\mathrm{clk}}$. They also differ in terms of maximum transmission rates on wired and wireless links ($R_{\mathrm{w}}$ and $R_{\mathrm{wl}}$, respectively). The availability of particular wireless link transmission rates is further limited by capabilities of built-in modems. Selected construction parameters of the TNC controllers are collected in Table 1.

**Table 1.** Construction parameters of TNC controllers used in tests

| Controller | Producer | Microprocessor | $f_{\mathrm{clk}}$ [MHz] | $R_{\mathrm{w}}$ [kbps] | $R_{\mathrm{wl}}$ [kbps] |
|---|---|---|---|---|---|
| TNC2 | (*prototype*) | Z80 | 2.4576 | 1.2–9.6 | 0.3–1.2 |
| TNC2D | Muel | Z80 | 4.9152 | 1.2–19.2 | 0.3–1.2 |
| Spirit-2 Standard | Paccomm | Z80 | 9.8304 | 4.8–57.6 | 4.8–57.6 |
| Spirit-2 High Speed | Paccomm | Z80 | 19.6608 | 4.8–57.6 | 4.8–57.6 |
| TNC3S | Symek | 68302 | 14.7456 | 1.2–115.2 | 1.2–614.4 |
| TNC7multi | Nt-G | LPC2106 | 58.9824 | 1.2–921.6 | 1.2–102.2 |

The controllers ran under the control of several types of software, namely:

- MFJ (initials of the author, Martin F. Jue) software (1.1.4, 1.1.9 and Muel versions), as delivered with TNC2, TNC2D and similar TNC2H controllers;
- TF (*The Firmware*) software (2.1d, 2.3b and 2.7b versions, all in 10 connections variant), as delivered with TNC2D and similar TNC2H controllers;
- Spirit-2 TNC software (5.0 version), as delivered with Spirit-2 controllers.

As all the aforementioned controllers are compatible with each other, the software can be easily interchanged by EPROM memory replacement.

The controllers were connected with a serial port to PC-class computers, as shown on Fig. 1. The connection between TNC's was also wired in order to avoid any radio interference that might influence on transmission process. Such

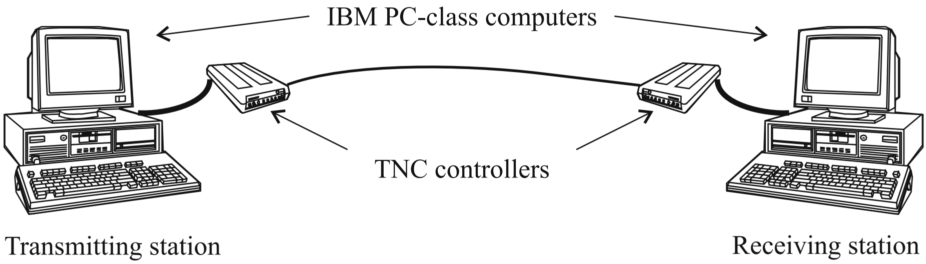



**Fig. 1.** Experimental network configuration

a connection does not violate time parameters of the network, because radio transceiver is always fully controlled by TNC.

The tested controllers acted as transmitters or receivers. In both cases, tested controller was connected in pair with either TNC3 or TNC7 controller. They are built using 16- or 32-bit microprocessors, belonging to the Motorola 68000 and LPC2000 (ARM7) families, respectively. During experimental tests, it was found out that these controllers achieve throughput close to the theoretical results [7], thus they should not be a bottleneck when connected with any Z80-based TNC.

During the tests, an 8 KB file was transmitted. The size was chosen as a compromise between transmission time and measurement accuracy. AX.25 protocol was configured for maximum theoretical throughput (window size $k = 7$, data field capacity $N_1 = 256$ bytes). Transmission time was measured from transmission start at the sender side to transmission end at the recipient. Although it takes into account transmission between computer and TNC, the influence of these times is negligible when total transmission time is sufficiently long [5]. Because of transmission rate ranges of TNC built-in modems (see Table 1), tests had to be performed separately using different controllers.

### 2.1 Results for "Slower" Configuration

In the "slower" configuration, TNC2 and TNC2D controllers were used with $R_{\mathrm{wl}} = 1.2$ kbps and $R_{\mathrm{w}} = 9.6$ kbps. Measurements results for TNC's acting as a sender or recipient are presented on Fig. 2 and Fig. 3, respectively. For comparison, the graphs contain also curves representing theoretical throughput of AX.25 protocol with immediate acknowledge generation (AX.25) or with $T_2$ delay (AX.25 T2), calculated according to [4].

In the case of tested TNC acting as a sender, we can see clearly that the effective throughput depends on the software used. All tested versions of MFJ and Spirit-2 software are very close to each other (to increase clarity, only Spirit curve is on the graph). Effective transmission speed grows rapidly when window size ($k$) increases from 1 to 4. However, further increasing of $k$ does not bring significant improvement of transmission speed. Probably there are some limitations in the software, e.g., buffer capacity, that does not allow to transmit, in average,

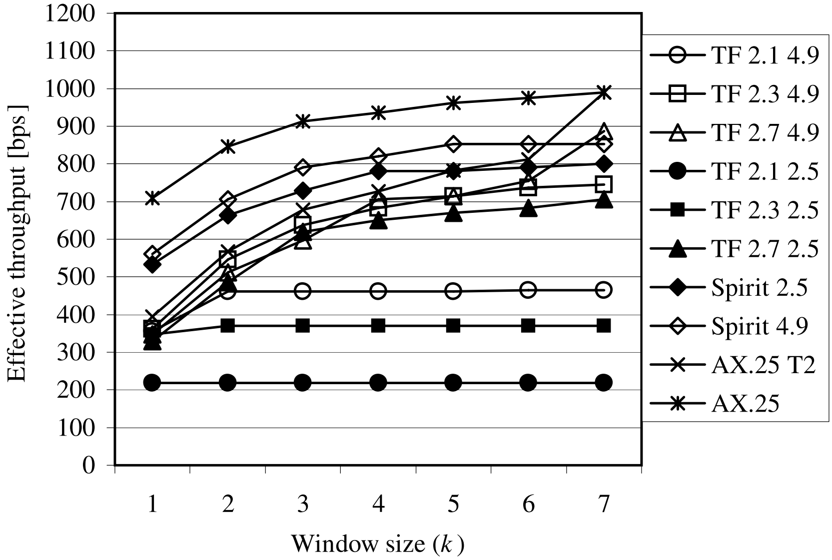



**Fig. 2.** Effective throughput for TNC2 (2.5 MHz) and TNC2D (4.9 MHz) as a sender

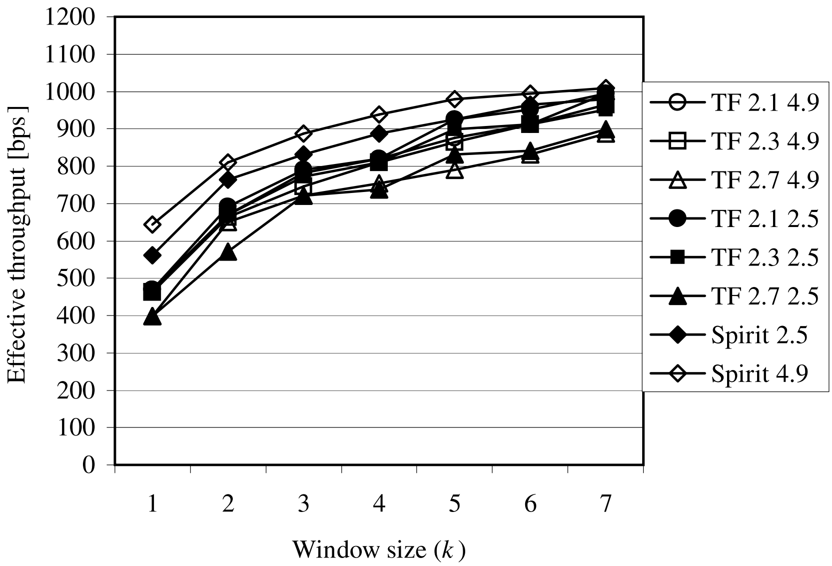



**Fig. 3.** Effective throughput for TNC2 (2.5 MHz) and TNC2D (4.9 MHz) as a recipient

more than 4 maximum-length frames. Maximum achievable effective transmission speed varies from about 750 to 800 bps for slower TNC and from 800 to 900 bps for the faster one. Both results are visibly below theoretical throughput of AX.25 protocol of about 1000 bps. TF software behaves completely different. Version 2.1 seems the most ineffective in the role of the sender and the achievable transmission speed is independent of window size. Similar is version 2.3, however, only when run on the slower TNC2; on the faster one, it behaves similarly to version 2.7. Thus, one might conclude that it requires more processing power than TNC2 can offer. Version 2.7 is the best one regardless of TNC speed. However, when run on the faster TNC2D, when $k = 7$, it achieves the same results as MFJ software.

When the tested TNC acts as a recipient, the difference between the fastest and the slowest software is much smaller. It can be found out, however, that the fastest reception proceeds under control of Spirit software, especially when run on a faster TNC2D. The slowest reception is for TF 2.7, regardless of microprocessor clock. It is also worth notice that these results are much closer to the theoretical throughput and vary from 900 to 1000 bps regardless of TNC clock frequency. Possible reason of this behavior is such that much faster TNC3 or TNC7 acts as a sender. It allows conclude that the sender processing power is much more important from the point of view of protocol efficiency than that of the recipient. Nevertheless, recipient software has still come influence on effective throughput.

### 2.2 Results for "Faster" Configuration

In the "faster" configuration, Spirit-2 in Standard and High Speed versions were used with $R_{\mathrm{wl}} = 9.6$ kbps and $R_{\mathrm{w}} = 57.6$ kbps. Measurements results for TNC's acting as a sender or recipient are presented on Fig. 4 and 5, respectively. The

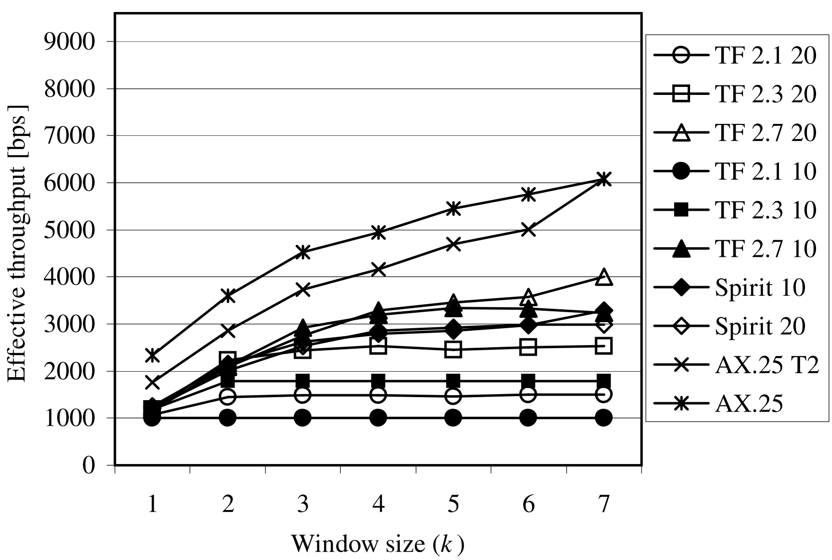



**Fig. 4.** Effective throughput for Spirit Standard (10 MHz) and High Speed (20 MHz) as a sender

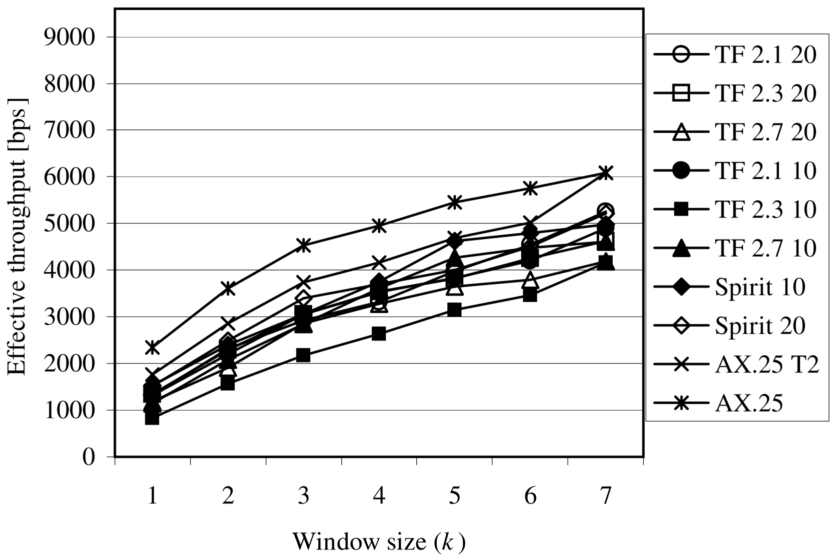



**Fig. 5.** Effective throughput for Spirit Standard (10 MHz) and High Speed (20 MHz) as a recipient

graphs contain also curves representing theoretical throughput of AX.25 protocol with immediate acknowledge generation (AX.25) or with $T_2$ delay (AX.25 T2).

If the tested TNC acts as a sender, the results are somewhat similar to those obtained in the slower configuration. Again, the slowest sender is TF 2.1, regardless of microprocessor clock frequency. a little bit better is TF 2.3. Both versions work faster at the faster microprocessor. However, they are both slower than Spirit and MFJ software in any version. These programs achieve similar efficiency regardless of clock frequency. This may lead to the conclusion that the transmission procedures are well optimised, however, some limitations exist in the software that does not allow reach even higher efficiency. It is especially visible for $k \geq 4$, where – similarly to the slower configuration – increasing windows size does not bring visible improvement in effective throughput. TF 2.7 achieves speeds similar to MFJ and Spirit. Nevertheless, when microprocessor works with faster clock (20 MHz), it outperforms all other software types, although not significantly. Maximum effective throughput measured in this test is about 4000 bps, while theoretically it could be as high as about 6000 bps. Thus, one can conclude that Z80-based TNC controllers are too slow to obtain performance that is high enough to use 9600 bps radio link efficiently, or the software – especially TF 2.7 – has not been sufficiently optimised.

If the tested TNC acts as a recipient, the results are also similar to the respective ones obtained in the slower configuration. The difference between the slowest and the fastest recipient are not very big – effective throughput ranges from about 4100 bps to about 5200 bps. Both results are visibly below theoretical estimation, which differs from the slower configuration where some measurement results were comparable to the calculated limit. The fastest recipients are TF 2.1 and Spirit, both at 20 MHz clock frequency. What seems surprising, TF 2.7 performs better for slower clock than for the faster one.

## 2.3 Real Window Size

In order to find out a more detailed reason of difference in effective throughput achieved for various software types and versions, we collected transmission log in a monitoring mode. Browsing this log allows analyse real frame exchange process. From the data gathered this way, we can obtain real window size distribution. Results for both slower and faster configurations are presented together on Fig. 6.

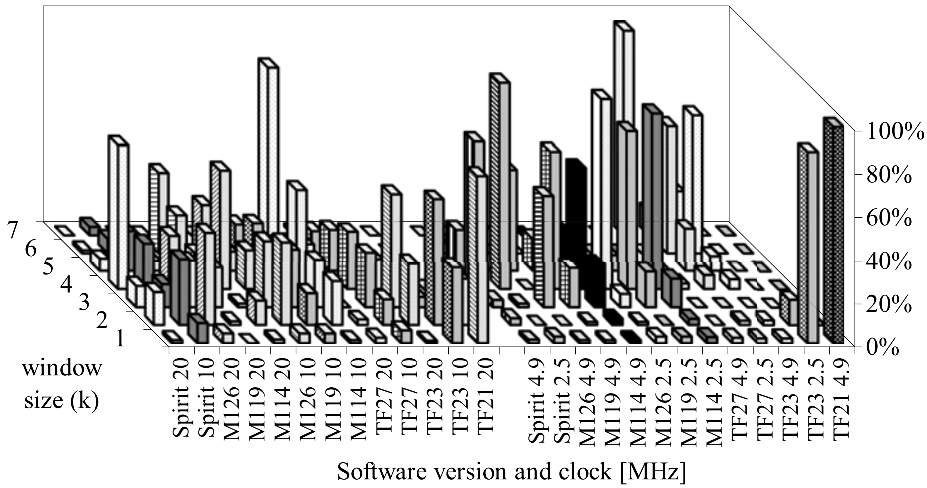



**Fig. 6.** Real window size for all tested TNC's and software versions

On the presented histograms, it can be easily seen that TF software behaves completely different than MFJ and Spirit ones. For slower configuration and MFJ or Spirit software, real window size oscillates around a value of 4. During transmission, depending on version, there is either always window size of 4, or 3 and 5 interleaving. For faster configuration, dominance of $k = 4$ is not that obvious. Indeed, smaller window sizes occur during transmission. TF software behaviour, in turn, depends more on clock frequency. If it is too low – say, 2.5 MHz – TF 2.1 and 2.3 use window size of 1 or 2. The situation gets better for TF 2.3 when clock is at least 4.9 MHz. However, TF 2.7 is even more effective and can use window size of 7. Unfortunately, it does not result in improvement of effective throughput, probably because it needs more time to process required amount of information. Surprisingly, when TF 2.1 or 2.3 operate on 10 or 20 MHz Z80, they use larger window size despite higher transmission rates. TF 2.7, running at 20 MHz, can still use window size of 7.

## 3 Conclusions

Presented results show that AX.25 protocol performance clearly depends on software type and version, as well as on transmission hardware processing power. However, software reaction for changing the hardware processing power differs. This leads to the conclusion that some TNC control programs (e.g., Spirit and all versions of MFJ) have some limitations that do not allow use maximum window size. On the other hand, TF software – especially in 2.7 version – can use maximum window size, but needs more processing power to process required amount of information. Thus, sometimes it shows worse performance than Spirit or MFJ software.

There are some protocol implementation issues that influences on its performance. Some software types mark the last frame in a window by P/F bit, some do not[1]. When a window size is less than 7, P/F bit requires immediate acknowledge; otherwise, the recipient waits for $T_2$ time in case there are more frames to be acknowledged together. Depending on software, $T_2$ time can be set manually or automatically according to $R_{\mathrm{wl}}$. The worst case occurs when the sender does not use P/F bit for last frame mark and the recipient waits for automatically set $T_2$. In this case, much time is lost during unnecessary waiting for more frames, thus reducing effective throughput regardless of other factors.

## References


1. Beech, W.A., Nielsen, D.E., Taylor, J.: AX.25 Link Access Protocol for Amateur Packet Radio. Tucson Amateur Packet Radio Corporation, Tucson (1997)
2. Zieliński, B.: Wireless computer networks using protocol conversion. PhD thesis, Silesian University of Technology, Gliwice, Poland (1998) (in Polish)
3. Wade, I. (ed.): Automatic Position Reporting System. APRS Protocol Reference. Protocol Version 1.0. Tucson Amateur Packet Radio Corporation, Tucson (2000)
4. Zieliński, B.: Efficiency estimation of AX.25 protocol. Theoretical and Applied Informatics 20(3), 199–214 (2008)
5. Zieliński, B.: An analytical model of TNC controller. Theoretical and Applied Informatics 20(4) (2008) (in press)
6. Dąbrowski, K.: Digital Amateur Communications. PWN, Warsaw (1994) (in Polish)
7. Zieliński, B.: A comparison of various types of TNC controllers. In: Slanina, Z., Srovnal, V. (eds.) Preprints of IFAC Workshop on Programmable Devices and Embedded Systems PDeS 2009, Rožnov pod Radhoštěm, pp. 80–85 (2009)


[1] It is neither required nor forbidden by the AX.25 protocol specification.

# Buffer Capacity Adjustment for TNC Controller

Bartłomiej Zieliński

Silesian University of Technology, Institute of Computer Science,
ul. Akademicka 16, 44-100 Gliwice, Poland
Bartlomiej.Zielinski@polsl.pl

**Abstract.** The paper presents an analytical method of buffer capacity adjustment for TNC controller that may be regarded as a network interface for amateur Packet Radio network. We consider the case of minimum buffer capacity that allows for continuous transmission at the sender side, regardless of transmission rates and effective throughput on wired and wireless link of TNC.

## 1 Introduction

Amateur Packet Radio network [1] may be considered as an example of a simple, wireless wide area network [2]. It was developed in early 1980's, when other connectivity media, such as Internet or cellular telephony, were not widely available yet. Later, despite its interesting properties, Packet Radio – as a deed of radio amateurs – did not achieve high popularity, which can be acknowledged by small quantity of widely available literature on this subject. Nevertheless, radio amateurs could use it for human-to-human communications similarly to current internet instant messengers. Nowadays, because of high popularity and availability of Internet and cellular telephony, application of Packet Radio changes for telemetry and remote control, mostly according to the requirements of APRS (*Automatic Position Reporting System*) protocol [3].

A complete Packet Radio station consists of a computer (or other DTE-type device) and radio transceiver. Because of different methods of information transmission, these devices can not cooperate with each other directly. It is thus necessary to apply specific data format processing techniques. Such a processing may be performed entirely in the computer (at a cost of higher processing power consumption) or by attachment of additional, external circuits dedicated for this application. An example of such a circuit is a TNC (*Terminal Node Controller*) [1].

## 2 TNC Controllers and AX.25 Protocol

TNC (*Terminal Node Controller*) [1] is an autonomous, microprocessor-based device used in amateur Packet Radio network. Its main purpose is a connection between a personal computer (or another DTE-type device, such as programmable controller or weather station) and radio receiver-transmitter operating in Packet Radio network. A typical Packet Radio network station with TNC controller is shown on Fig. 1.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 119–126, 2009.

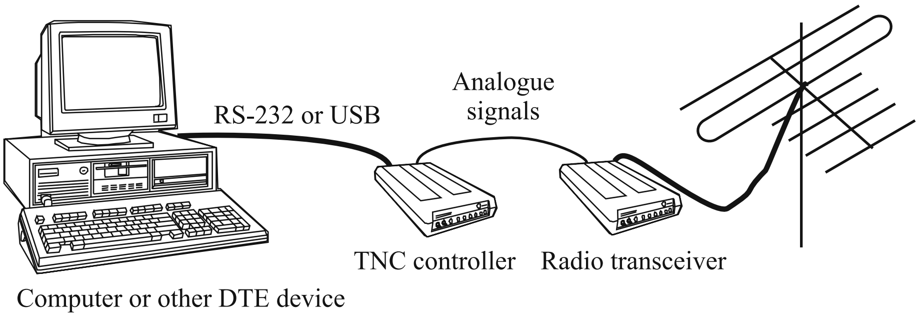



**Fig. 1.** Typical Packet Radio station with TNC controller

## 2.1 TNC Controller Structure

TNC controller consists of a digital part, that ensures data format processing according to Packet Radio network requirements and rules of AX.25 protocol, as well as analogue part, that plays a function of a modem and makes it possible to control radio transceiver directly from TNC. The block diagram of TNC hardware is presented on Fig. 2. A more detailed description of properties of individual types of TNC controllers may be found in [4].

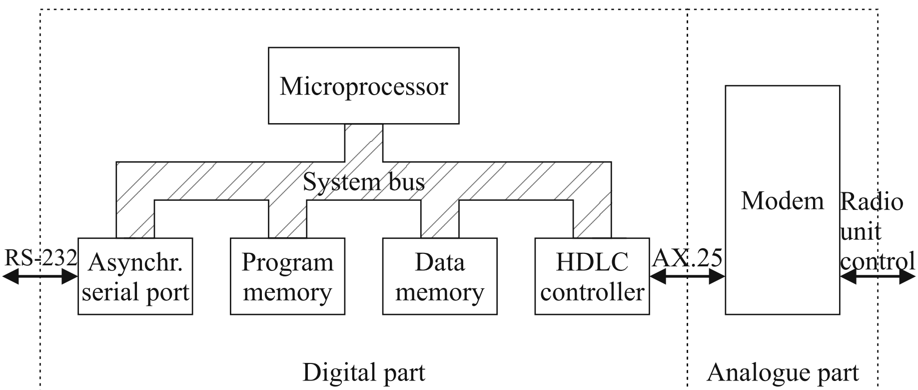



**Fig. 2.** Block diagram of TNC controller hardware

## 2.2 TNC Controller Functions

The fundamental task of TNC controller is to process format of data incoming from attached computer so that it could meet requirements of the AX.25 protocol, used as a data link layer in Packet Radio network. With this end in view, the controller buffers data, and then places them in adequately formed frames. During sending on radio link, a modulation is used according to the selected transmission rate. It allows for direct connection to radio transceiver and control over its work from TNC. We can therefore say that TNC controller is responsible for proper realisation of AX.25 protocol; however, it does not have radio communication capabilities itself. Thanks to such approach, TNC controllers may be used with any radio transceiver, not only in radio amateur frequency bands.

Functions described above are controlled by a software. It is responsible for proper realisation of network mechanisms, and furthermore, it contains a user interface that allows for, among others, configuration of some controller and radio link parameters, as well as management of logical links with other network stations. Depending on controller type, various methods of computer (user) to controller communication are available. They may be optimised for controller's cooperation with a human (TAPR and TF command sets) or a device (HOST and KISS modes). Availability of individual operating modes depends on controller type.

### 2.3 Parameters of AX.25 Protocol

AX.25 protocol and TNC controller behaviour may be adjusted by large number of parameters (depending on TNC software). From the point of view of protocol efficiency, the most important are the following parameters:

$k$ – windows size, i.e., maximum number of frames sent consecutively before waiting for an acknowledge; not larger than 127 in protocol version 2.2 and 7 in older versions;

$N_1$ – maximum capacity of data field in an information or unnumbered frame; not larger than 256;

$T_2$ – time that elapses between the end of latest I frame and acknowledge; depending on software and parameters, this delay may or may not occur during transmission and may be set manually or automatically;

$T_{102}$ – slot duration for carrier sense-based collision avoidance; typically about 50 to 300 ms;

$T_{103}$ – time for stabilisation of transmitter parameters after transmission start and for carrier detection in receiver; depends on radio transceiver capabilities and, in most cases, varies from few tens to few hundreds milliseconds;

$p$ – persistence parameter of carrier sense-based collision avoidance (not defined by protocol description but implemented in most of the software); typically equal to 63 which means transmission probability of 25%.

## 3 Buffer Capacity Adjustment

Let's assume that we have two computers (or other DTE devices, e.g., PLC controllers) connected with two TNC controllers. Such configuration is shown on Fig. 3.

In such a network – because of buffering of transmitted information in TNC memory and data processing – the transmission runs in three stages:

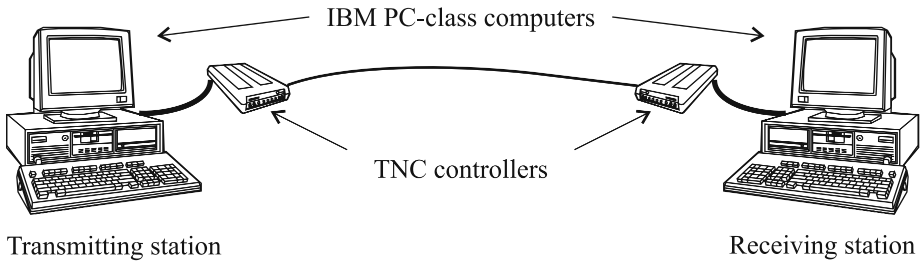



**Fig. 3.** Network configuration accepted for considerations

1. wired transmission between the sender and transmitting TNC;
2. wireless transmission between TNC's;
3. wired transmission between the receiving TNC and the recipient.

These stages are shown on Fig. 4.

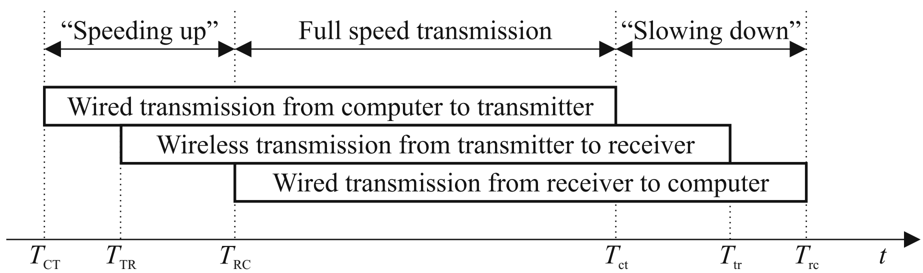



**Fig. 4.** Transmission stages using TNC controllers

In most cases, the effective throughput of the wireless link is much lower than that of the wired link. Thus, data arriving to the TNC during first transmission stage must be placed in a buffer before it can be processed, prepared for wireless transmission and finally sent. However, when the buffer capacity is reached, the first transmission stage should stop in order to avoid data loss due to buffer overflow. This can be done by either hardware or software flow control on RS-232 link, depending on availability of particular signals in a connector.

On the other hand, certain applications may require continuous transmission at the sender side. In order to make it possible, we must provide buffer that is large enough to store the information awaiting to be sent. Obviously, its capacity depends on the effective throughput of the wired and wireless link. We predict that with increasing of the effective throughput of the wireless link, or with decreasing that of the wired link, required buffer capacity decreases.

### 3.1 Estimation of Effective Throughput

Calculation of the effective transmission speed of a wired link is straightforward. Assuming that RS-232 works at $R_\mathrm{w}$ bps and is configured so that each character is represented using 10 bits, we get:

$$V_\mathrm{w} = \frac{8R_\mathrm{w}}{10} \ . \tag{1}$$

The wireless link may be a half-duplex or a full-duplex one. According to the previously developed theoretical model of AX.25 protocol [5], for the half-duplex case, $L_\mathrm{D}$-bytes long file transmission takes

$$T_\mathrm{p} = \left\lceil \frac{L_\mathrm{D}}{kN_1} \right\rceil \left( \frac{256T_{102}}{2(p+1)} + 2T_{103} + T_2 + (1+k)\frac{63}{62}\frac{160}{R_\mathrm{wl}} \right) + \left\lceil \frac{L_\mathrm{D}}{N_1} \right\rceil \frac{63}{62}\frac{8N_1}{R_\mathrm{wl}} \ , \tag{2}$$

where $R_\mathrm{wl}$ – wireless link transmission rate [bps]. Similar time for the full-duplex link equals to

$$T_\mathrm{p} = T_{103} + \left\lceil \frac{L_\mathrm{D}}{N_1} \right\rceil \left( \frac{63}{62}\frac{160+8N_1}{R_\mathrm{wl}} \right) + \left( \frac{63}{62}\frac{160}{R_\mathrm{wl}} \right) \ . \tag{3}$$

In both cases, we can calculate effective transmission speed for wireless link by dividing data size in bits by $T_\mathrm{p}$:

$$V_\mathrm{wl} = \frac{8L_\mathrm{D}}{T_\mathrm{p}} \ . \tag{4}$$

### 3.2 Estimation of Buffer Capacity

Minimum buffer capacity that ensures continuous transmission at the sender side depends on total size of transmitted data as well as on difference between effective transmission speeds of wired and wireless links. We may assume that effective transmission speed of wired link on the sender side corresponds to average speed of buffer filling. In turn, effective speed of radio link corresponds to average speed of buffer emptying. Multiplying the difference between them by the time of data transmission on wired link we may calculate approximate buffer size that guarantees continuous transmission on wired link [6]:

$$C = (V_\mathrm{w} - V_\mathrm{wl})(T_\mathrm{ct} - T_\mathrm{CT}) = \left( \frac{8R_\mathrm{w}}{10} - \frac{8L_\mathrm{D}}{T_\mathrm{p}} \right) \frac{10L_\mathrm{D}}{8R_\mathrm{w}} = L_\mathrm{D}\left( 1 - \frac{10L_\mathrm{D}}{R_\mathrm{w}T_\mathrm{p}} \right) \ . \tag{5}$$

It is also possible to ensure continuous transmission on the receiver side, however, it requires that entire data size ($L_\mathrm{D}$) is known by receiving TNC controller. In this case, additional delay of transmission start at the receiver side ($T_\mathrm{RC}$ as shown on Fig. 4) is necessary. We must also take into account that the data transmission runs from the wireless link to the wired one. Thus, average speed of buffer filling corresponds to the effective speed of the wireless link, while average speed of buffer emptying – to that of wired link.

In practice, continuous transmission on the recipient side requires that entire data is already received by the receiving TNC controller. Otherwise, due to possible transmission errors, data might not be delivered in time to ensure continuous transmission on recipient's wired link.

### 3.3 Calculation Results

According to (5), we can calculate the dependency between minimum buffer size that assures continuous transmission at the sender side and transmission rates for both wired and wireless links. As the effective throughput of AX.25 protocol differs on radio link variant (half- or full-duplex), the estimated buffer size is expected to depend on link variant, too. Substituting (2) or (3) for $T_p$ in (5), we can get results for half-duplex or full-duplex radio link, respectively. Parameters accepted for calculations are: $L_D = 16384$ bytes, $k = 7$, $N_1 = 256$ bytes. Calculation results are presented on Fig. 5 and 6, respectively.

In the case of half-duplex link, we can see that the buffer capacity grows rapidly and reaches almost $L_D$ regardless of wireless link transmission rate. For example, when $R_{wl} = 1.2$ kbps and $R_w = 2.4$ kbps, required buffer capacity reaches almost 8 KB. It means that a half of transmitted information must wait to be sent. When $R_w = 4.8$ kbps, capacity must be at least 12 KB, when $R_w = 9.6$ kbps – 14 KB. For the highest wireless link transmission rate considered, i.e., $R_{wl} = 614.4$ kbps, required buffer capacity is about 8 KB when $R_w = 38.4$ kbps. It can be easily understood, because effective throughput for

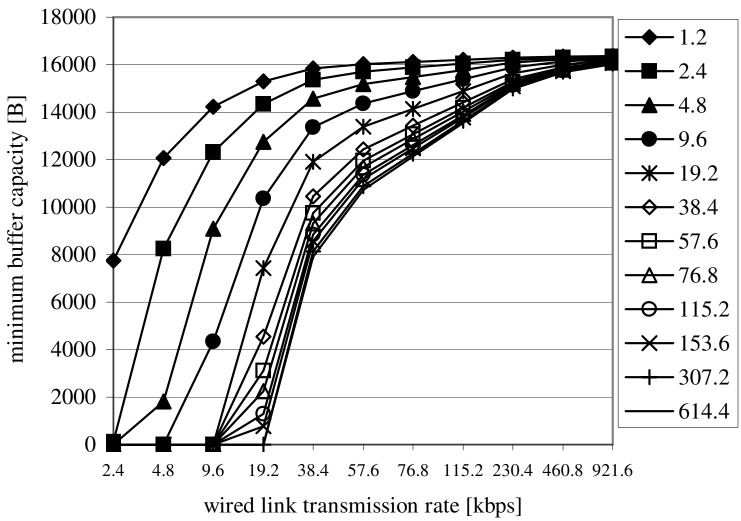



**Fig. 5.** Minimum buffer size for half-duplex radio link

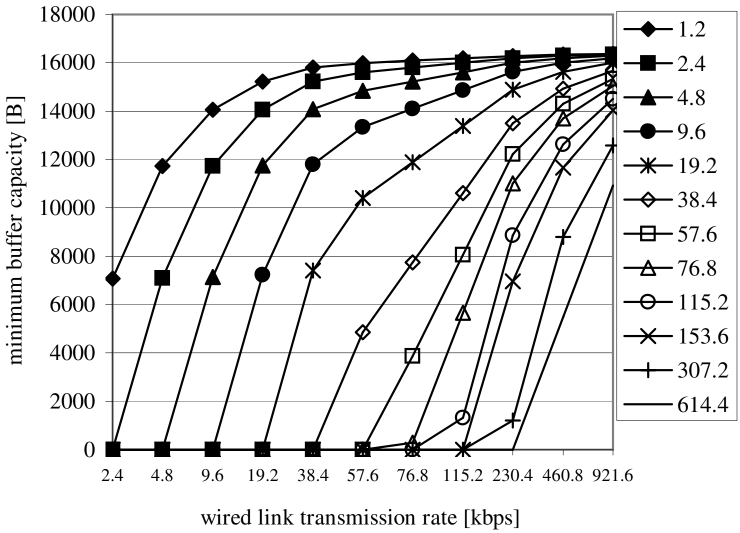



**Fig. 6.** Minimum buffer size for full-duplex radio link

this $R_{\mathrm{wl}}$ is only about 28 kbps [5]. In general, the higher the wireless link transmission rate ($R_{\mathrm{wl}}$) is, or the lower the wired link transmission rate ($R_{\mathrm{w}}$) is, the later the rapid grow begins. This phenomenon reflects the difference between the effective throughput of wired and wireless links. Thus, the wired link effective speed – which depends on transmission rate only – may exceed wireless link effective speed easily.

In the case of full-duplex link, the required buffer capacity grows almost as fast as for half-duplex one. However, the grow begins at higher transmission rates of wired link. For example, buffer capacity for $R_{\mathrm{wl}} = 1.2$ kbps is almost as large as for the half-duplex link, practically regardless of wired link transmission rate. On the other hand, when $R_{\mathrm{wl}} = 614.4$ kbps, the buffer is not required for $R_{\mathrm{w}}$ up to 230.4 kbps. For higher rates, the minimum buffer capacity grows and for $R_{\mathrm{w}} = 921.6$ kbps it reaches about 11 KB. For $R_{\mathrm{w}} = 460.8$ kbps, it is about 5.5 KB, which is a bit surprising, because effective throughput of wireless link for this parameter set is about 560 kbps that is more than that of wired link; in this case, no buffering should be necessary.

## 4 Conclusions

Presented results, achieved according to derived analytical equations, show that there is a dependency between transmission rates of both wired and wireless link and minimum buffer size that allows for continuous transmission at the sender side. As expected, buffer size increases when either wired link transmission rate grows or the effective speed of wireless link falls. It is worth emphasize that the wireless link properties depend on several transmission parameters, hence effective throughput must be considered instead of transmission rate. This phenomenon is especially visible for half-duplex link.

Unfortunately, presented theoretical estimations cannot be easily verified. During practical tests with TNC controllers [4] it was shown, that even for a relatively small file (few KB size) transmission on RS-232 link was stopped by TNC in order to prevent buffer overflow, even if a controller was equipped with much more memory (e.g., more than 100 KB). This leads to the conclusion that only a part of RAM installed in TNC is used to buffer the data awaiting to be sent. A more detailed analysis requires an access to the source code of TNC control software. However, even without the source code, one can see that TNC controllers utilize their memory in a different way than proposed in the paper. Thus, possible modification of the software allows verify presented results in practise.

It is worth notice, however, that TNC controller is a good model of a protocol converter that allows integrate wired and wireless network segments [7]. From this point of view, presented considerations may be useful as an example during design of such a converter for given network protocols.

## References

1. Dąbrowski, K.: Digital Amateur Communications. PWN, Warsaw (1994) (in Polish)
2. Karn, P.R., Price, H.E., Diersing, R.J.: Packet radio in the amateur service 3(3), 431–439 (1985)
3. Wade, I. (ed.): Automatic Position Reporting System. APRS Protocol Reference. Protocol Version 1.0. Tucson Amateur Packet Radio Corporation, Tucson (2000)
4. Zieliński, B.: A comparison of various types of TNC controllers. In: Slanina, Z., Srovnal, V. (eds.) Preprints of IFAC Workshop on Programmable Devices and Embedded Systems PDeS 2009, Rožnov pod Radhoštěm, pp. 80–85 (2009)
5. Zieliński, B.: Efficiency estimation of AX.25 protocol. Theoretical and Applied Informatics 20(3), 199–214 (2008)
6. Zieliński, B.: An analytical model of TNC controller. Theoretical and Applied Informatics 20(4) (in press) (2008)
7. Zieliński, B.: Wireless computer networks using protocol conversion. PhD thesis, Silesian University of Technology, Gliwice, Poland (1998) (in Polish)

# The Reliability of Any-Hop Star Networks with Respect to Failures of Communication Nodes

Jadwiga Kozłowska

Gdańsk University of Technology,
Faculty of Electronics, Telecommunications and Informatics

**Abstract.** This paper investigated the reliability of any-hop star networks. The any-hop star topology is used in centralized computer networks. All network nodes fail independently, links are failure-free. Following measures of network reliability are assumed: the expected number of nodes which can communicate with the central node; the expected number of node pairs which are connected by a path through the central node; the expected number of node pairs communicating. Any two nodes can communicate, if links and nodes forming routes between these nodes have not failed. The values of reliability measures were counted for one-, two-, three-, and four-hop star, as well as chain networks. Obtained results compared with reliability of the networks with unreliable links.

## 1 Introduction

The reliability of network depends strongly on their topology and on the reliability of the communication links and nodes. The following measures of the reliability are assumed: the expected number of nodes which can communicate with the central node; the expected number of node pairs which are connected by a path through the central node; the expected number of node pairs communicating. Any two nodes can communicate, if links and nodes forming routes as between these nodes have not failed. There are many well-known methods [1], [6] of the reliability analysis. Almost all these methods used a set of branch or node-disjoint routes, or cuts. For any network, calculation of all disjoint routes or cut set can be difficult. In the case when topology of network is the tree or star the problem is simpler. For these networks expected number of nodes which can communicate with the central node is equal to the sum of probabilities of reliability routes connecting nodes with central node. Similarly, the expected number of node pairs which are connected by a path through the central node, and the expected number of node pairs communicating are equal to sum probabilities of reliability routes connecting node pairs. For large networks with many hops the recurrence method [2] can be used instead of determining the values of the reliability measures. The calculations of the network reliability were done as a function of the probability of node being operative. The calculations of the network reliability as a function of the probability of link being operative were done in the papers [5]. In the paper [3], [4] the numerical calculations for assumed measures of reliability were done for the star, ring and hybrid networks.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 127–134, 2009.

## 2 Reliability Analysis

The topology of the computer communication network will be modelled as a graph $G = (N, E)$ where $N = \{n_1, n_2, \ldots, n_N\}$ is a set of nodes, $E$ is a set of links. Considered networks have any-hop star topologies. Example of two-hop star topology is shown in Fig. 1. All of the nodes fail independently, links are failure-free. The probability that the node $n_i$ is in the non-damaged state is equal to $p(i)$. The following measures of the reliability are considered: the expected number of nodes that can communicate with the central node($S$); the expected number of node pairs that are connected by a path through the central node($R$); the expected number of node pairs communicating ($T$).

The idea of the recurrence method is the following. Suppose, that for each node $i$ except central node (node 1) is designed a node $F(i)$, such that $F(i) < i$ and $(i, F(i))$ is a link in the network. In Fig. 2 the network contains two subtrees, one with the central node $i$ and the other with the central node $j = F(i)$. The values of the measures of reliability for subtrees are known.

The values of reliability measures for the tree obtained by joining $i$ and $j$ by link $(i, j)$ determine recurrence relations (Eq. 1) [2], where $S(i)$ $(S(j))$ is the expected number of nodes in the subtree which communicate with the central node $i$ $(j)$, including $i$ $(j)$, $R(i)(R(j))$ the expected number of node pairs in the subtree communicating through central node, $T(i)(T(j))$ the expected number of node pairs communicating in the subtree, $R(j)^{'}, T(j)^{'}, S(j)^{'}$ are the values of the reliability measures for the tree obtained by joining node $i$ and node $j$ by link $(i, j)$.

$$\begin{aligned} R(j)^{'} &\leftarrow R(j) + S(i)S(j) + R(i)p(j) \\ T(j)^{'} &\leftarrow T(i) + T(j) + S(i)S(j) \\ S(j)^{'} &\leftarrow S(j) + S(i)p(j) \end{aligned} \tag{1}$$

Algorithm is following [2]:

Step 0: (Initialization) Set $R(i) = 0$, $S(i) = p(i)$, $T(i) = 0$, $i = 1, 2, \ldots, N$, set $i = N$. Go to step 1.
Step 1: Let $j = F(i)$, and set $R(j)$ to $R(j) + S(i)S(j) + R(i)p(j)$, set $T(j)$ to $T(i) + T(j) + S(i)S(j)$, and then set $S(j)$ to $S(j) + S(i)p(j)$. Go to step 2.
Step 2: Set $i$ to $i - 1$. If $i = 1$, stop; otherwise, go to step 1.

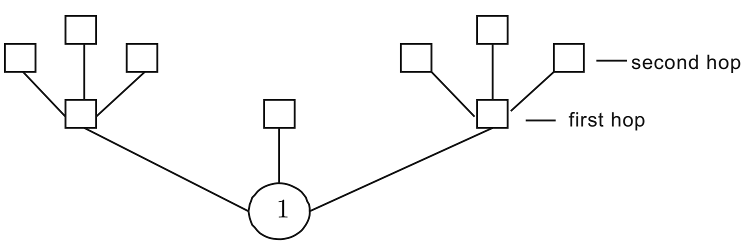



**Fig. 1.** Two-hop star topology

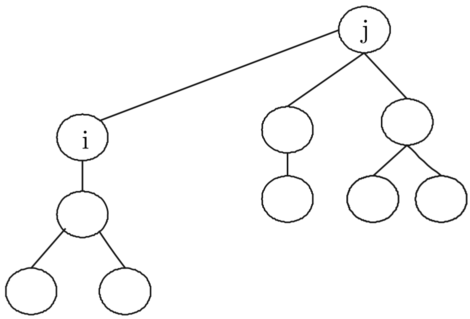



**Fig. 2.** Illustration for recurrence relations

When the algorithm stops $R(1)$ is the expected number of node pairs which are connected by a path through the central node, $S(1)$ is the expected number of nodes which can communicate with the central node, and $T(1)$ is the expected number of node pairs communicating. The obtained numerical results are presented in next section.

## 3 Numerical Results

The values of reliability measures were calculated for one, two, three, four and hundred- hop star networks. The nodes fail independently with the same probability $1-p$ and the links are failure-free. The values of reliability measures were counted for the networks containing hundred and one nodes. The distribution of

**Table 1.** Obtained values for the expected number of nodes which can communicate with the central node ($S$) (in percent) as a function of the probability of a node being operative

| $p$ | $SG1(101,p)$ one-hop star network | $SG2(101,p)$ two-hop star network | $SG3(101,p)$ three-hop star network | $SG4(101,p)$ four-hop star network | $SG100(101,p)$ hundred-hop star network |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0.1 | 1.089 | 0.287 | 0.159 | 0.137 | 0.11 |
| 0.2 | 4.158 | 1.307 | 0.585 | 0.416 | 0.248 |
| 0.3 | 9.208 | 3.594 | 1.648 | 1.045 | 0.424 |
| 0.4 | 16.238 | 7.683 | 3.919 | 2.448 | 0.66 |
| 0.5 | 25.248 | 14.109 | 8.168 | 5.353 | 0.99 |
| 0.6 | 36.238 | 23.406 | 15.365 | 10.877 | 1.485 |
| 0.7 | 49.208 | 36.109 | 26.678 | 20.613 | 2.31 |
| 0.8 | 64.158 | 52.752 | 43.476 | 36.718 | 3.96 |
| 0.9 | 81.089 | 73.871 | 67.327 | 62 | 8.911 |
| 1 | 100 | 100 | 100 | 100 | 100 |

**Table 2.** Obtained values for the expected number of node pairs which are connected by a path though the central node (R) (in percent) as a function of the probability of a node being operative

| $p$ | $RG1(101, p)$ one-hop star network | $RG2(101, p)$ two-hop star network | $RG3(101, p)$ three-hop star network | $RG4(101, p)$ four-hop star network | $RG100(101, p)$ hundred-hop star network |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0.1 | 0.118 | 0.009 | 0.002 | 0.001 | 0.0002 |
| 0.2 | 0.863 | 0.104 | 0.024 | 0.01 | 0.001 |
| 0.3 | 2.825 | 0.501 | 0.137 | 0.057 | 0.004 |
| 0.4 | 6.59 | 1.649 | 0.545 | 0.242 | 0.009 |
| 0.5 | 12.748 | 4.31 | 1.74 | 0.861 | 0.02 |
| 0.6 | 21.885 | 9.649 | 4.753 | 2.674 | 0.045 |
| 0.7 | 34.591 | 19.319 | 11.531 | 7.47 | 0.108 |
| 0.8 | 51.453 | 35.549 | 25.464 | 19.085 | 0.317 |
| 0.9 | 73.06 | 61.23 | 52.088 | 45.162 | 1.603 |
| 1 | 100 | 100 | 100 | 100 | 100 |

**Table 3.** Obtained values for the expected number of node pairs communicating (T) (in percent) as a function of the probability of a node being operative

| $p$ | $TG1(101, p)$ one-hop star network | $TG2(101, p)$ two-hop star network | $TG3(101, p)$ three-hop star network | $TG4(101, p)$ four-hop star network | $TG100(101, p)$ hundred-hop star network |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 |
| 0.1 | 0.118 | 0.032 | 0.028 | 0.025 | 0.022 |
| 0.2 | 0.863 | 0.206 | 0.154 | 0.129 | 0.099 |
| 0.3 | 2.825 | 0.748 | 0.485 | 0.382 | 0.254 |
| 0.4 | 6.59 | 2.094 | 1.255 | 0.936 | 0.525 |
| 0.5 | 12.748 | 4.978 | 2.954 | 2.124 | 0.98 |
| 0.6 | 21.885 | 10.521 | 6.559 | 4.708 | 1.755 |
| 0.7 | 34.591 | 20.314 | 13.877 | 10.357 | 3.159 |
| 0.8 | 51.453 | 36.507 | 28.028 | 22.554 | 6.083 |
| 0.9 | 73.06 | 61.894 | 54.099 | 48.166 | 14.596 |
| 1 | 100 | 100 | 100 | 100 | 100 |

the nodes is following. The two-hop star network contains ten nodes in the first hop, and each node of the first hop is connected with nine nodes of second hop. The three-hop star network contains four nodes in the first hop, each node of the first hop is connected with three nodes of the second hop, and each node of the second hop is connected with seven nodes of the third hop. The four-hop network contains three nodes in the first hop, each node of the first hop is connected with two nodes of the second hop, and each node of the second hop is connected with three nodes of the third hop, and each node of third hop is connected with four nodes of fourth-hop. The each node of hundred-hop star is connected with one

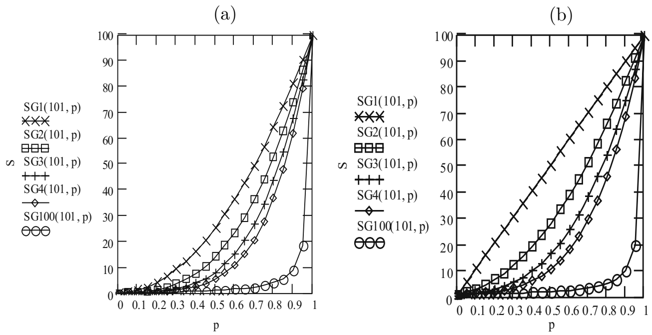



**Fig. 3.** The expected number of nodes which can communicate with the central node: (a) as a function of the probability of a node being operative; (b) as a function of the probability of a link being operative

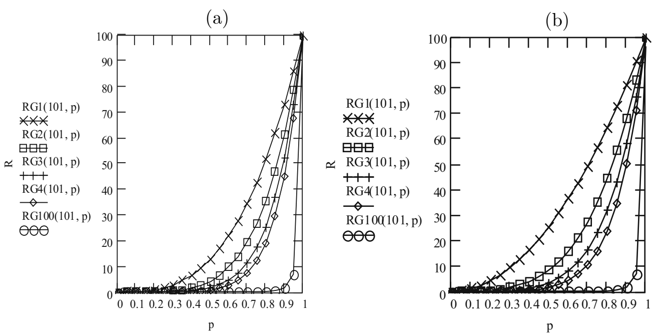



**Fig. 4.** The expected number of node pairs which are connected by a path through the central node: (a) as a function of the probability of a node being operative; (b) as a function of the probability of a link being operative

node next hop. The calculations were done as a function of the probability of a node being operative. The obtained results are shown in Table 1, 2, 3.

Figures 3, 4, 5 show differences between the reliability network with unreliable nodes and network with unreliable links [6].

Figures shown that the differences between networks' reliability decrease when the number of hops of the star network grows. The largest differences are for the probabilities of link and node failure equal to 0.3, 0.2, and 0.1. The differences for assumed measures are similar. For example, in the case when links are unreliable and nodes are reliable for criterion S the largest difference between one-hop and two-hop star network is for $p = 0.5$ and equals 22%. The largest difference between two-hop and three-hop star network is for $p = 0.7$ and equals 13%. In the case unreliable nodes, the largest difference between one-hop and two-hop star network is for $p = 0.7$ and equals 13%, the largest difference between two-hop and three-hop star network equals 9%. For criterion $R(T)$ the largest

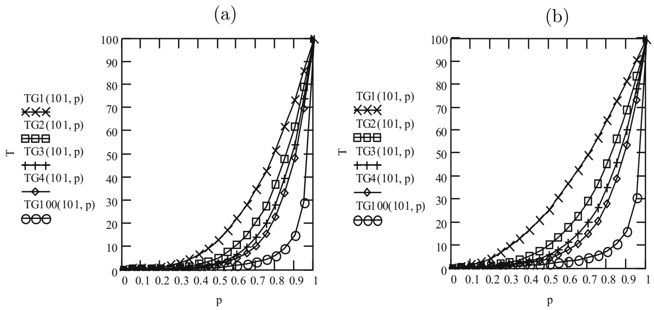



**Fig. 5.** The expected number of node pairs communicating: (a) as a function of the probability of a node being operative; (b) as a function of the probability of a link being operative

**Table 4.** Obtained values for the expected number of node pairs communicating (T) (in percent) and values for the expected number of nodes which can communicate with the central node (S) (in percent) as a function of the probability of a node being operative

| $p$ | $TG1(11,p)$ | $TG1(101,p)$ | $TG1(301,p)$ | $SG1(11,p)$ | $SG1(101,p)$ | $SG1(301,p)$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0.1 | 0.264 | 0.118 | 0.106 | 1.818 | 1.089 | 1.03 |
| 0.2 | 1.382 | 0.863 | 0.821 | 5.455 | 4.158 | 4.053 |
| 0.3 | 3.845 | 2.825 | 2.742 | 10.909 | 9.208 | 9.07 |
| 0.4 | 8.145 | 6.59 | 6.464 | 18.182 | 16.238 | 16.08 |
| 0.5 | 14.773 | 12.748 | 12.583 | 27.273 | 25.248 | 25.083 |
| 0.6 | 24.218 | 21.885 | 21.696 | 38.182 | 36.238 | 36.08 |
| 0.7 | 36.973 | 34.591 | 34.398 | 50.909 | 49.208 | 49.07 |
| 0.8 | 53.527 | 51.453 | 51.285 | 65.455 | 64.158 | 64.053 |
| 0.9 | 74.373 | 73.06 | 72.954 | 81.818 | 81.089 | 81.03 |
| 1 | 100 | 100 | 100 | 100 | 100 | 100 |

difference between one-hop and two-hop star network is for $p = 0.7$ and equals 22% (20%). The largest difference between two-hop and three-hop star network is for $p = 0.8$ and equals 12% (10%). In the case when nodes are unreliable and links are reliable for criterion $R(T)$ the largest difference between one-hop and two-hop star network is for $p = 0.8$ and equals 16% (15%), the largest difference between two-hop and three-hop star network equals 10% (8%).

Figures 3, 4, 5 show that the networks with unreliable links are more reliable than the networks with unreliable nodes. The differences between networks' reliability decrease when the number of hops of the star network grows. For the measure $S$, the largest difference for one-hop star network equals 25%, for two-hop star network equals 16%, for three-hop star network equals 12%, for hundred-hop star network equals 2%. For the measure $R$, the largest difference for one-hop star network equals 16%, for two-hop star network equals 9%, for three-hop star

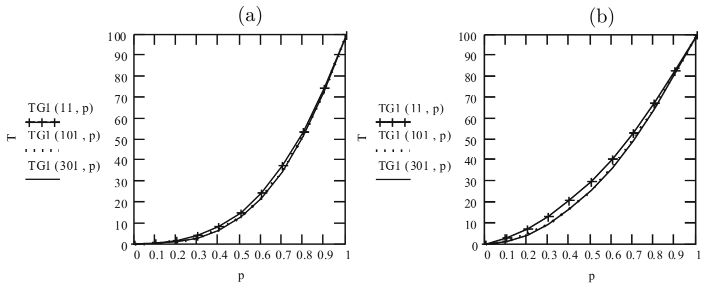



**Fig. 6.** The expected number of node pairs communicating for different number of the nodes: (a) as a function of the probability of a node being operative; (b) as a function of the probability of a link being operative

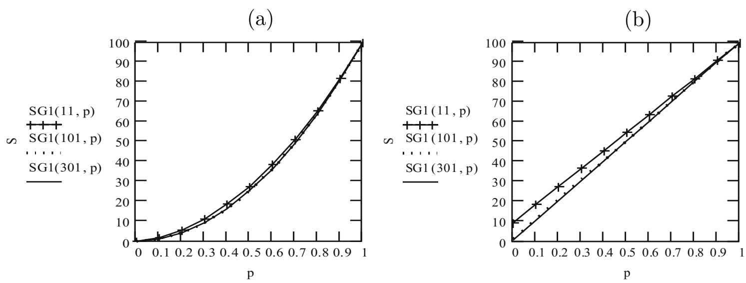



**Fig. 7.** The expected number of nodes, which can communicate with the central node for different number of the nodes: (a) as a function of the probability of a node being operative; (b) as a function of the probability of a link being operative

network equals 7%, for hundred-hop star network equals 0.1%. For the measure $T$, the largest difference for one-hop star network equals 15%, for two-hop star network equals 9%, for three-hop star network equals 7%, for hundred-hop star network equals 2%. The values of the reliability measures for one-hop star network were calculated for different number of nodes. The obtained results for eleven, hundred and one, and three hundred and one nodes are shown in Table 4. Figures 6 and 7 illustrate numerical results obtained for reliability measures.

Figures 6, 7 show that the differences between the networks' reliability decrease when number of nodes grows. The networks with unreliable links are more reliable than the networks with unreliable nodes. For the measure S, the largest difference equals 27%, for the measure $T$, the largest difference equals 16%.

## 4 Conclusion

This chapter investigated the reliability of any-hop star networks, whose nodes fail independently. Several measures of reliability have been considered. The recurrence method is used in analyzing networks. The values of reliability measures were calculated for one, two, three, four and hundred- hop star networks. The numerical calculations the network reliability measures were considered as a function of the probability of a node failures. Obtained results show that the differences between networks' reliability decrease when the number of hops of the star network grows. The largest differences are for the probabilities of link and node failure equal to 0.3, 0.2, and 0.1. Obtained results compared with reliability of the networks with unreliable links. The networks with unreliable links are more reliable than the networks with unreliable nodes. The differences between them decrease when the hop number and number of nodes of the networks grow.

## References

1. Hansler, E.: Computational aspects of network reliability problems, Presented at the NATO Advanced Study Institute New Concept in Multi-User Communications. Norwick, U.K 4-6 (1980)
2. Kershenbaum, A., Van Slyke, R.M.: Recursive analysis of network reliability. Networks (3), 81–94 (1973)
3. Kozłowska, J.: Niezawodność sieci komputerowych o wielostopniowych strukturach pętlowych, gwiaździstych oraz mieszanych. In: Węgrzyn, S., et al. (eds.) Nowe technologie sieci komputerowych (in Polish), vol. 1, pp. 165–172. WKŁ, Warszawa (2006)
4. Kozłowska, J.: Niezawodność scentralizowanych sieci komputerowych o dwustopniowych strukturach gwiaździstych oraz mieszanych. In: Kwiecień, A., et al. (eds.) Sieci komputerowe. Aplikacje i zastosowania (in Polish), vol. 2, pp. 253–261. WKŁ, Warszawa (2007)
5. Kozłowska, J.: The reliability of tree and star networks. In: Kwiecień, A., et al. (eds.) Contemporary aspect of computer networks, vol. 2, pp. 147–156. WKŁ, Warszawa (2008)
6. Zabłudowski, A.: A recursive method for network reliability measures evaluation. Microelectron. Reliab. 24(3), 445–451 (1984)

# A Floor Description Language as a Tool in the Process of Wireless Network Design

Remigiusz Olejnik*

West Pomeranian University of Technology, Szczecin,
Faculty of Computer Science and Information Systems,
ul. Zolnierska 49, 71-210 Szczecin, Poland
r.olejnik@ieee.org

**Abstract.** The paper describes a Floor Description Language (FDL) which could be used as a key chain in the process of wireless network design. Traditional ways of designing wireless networks are also presented. Numerical results and comparison with commercial systems summarizes the paper and proves FDL's usability.

## 1 Introduction

Currently density of modern wireless networks grows very fast. Many of the deployments are not sufficiently prepared – most of home and small office wireless networks have never been designed by a designer or network engineer – just users bought network equipment, turned it on, connected to the rest of their network and used it "as is". Such approach is very inefficient in terms of the quality indices such as overall network set-up costs, maximal network range and throuhput. This paper presents a novel Floor Description Language (FDL) which along with well known propagation model makes designing of the wireless networks much easier with ready map of the floor.

The paper consists of three parts. First part describes known ways of wireless network designing and propagation models. Second part presents novel Floor Description Language used in the process of network design. Third part shows numerical results and comparison with commercial systems. Summary containing possible future works ends the paper.

## 2 Wireless Network Design Process

Traditional process of network design is composed of four main steps:

1. gathering necessary data on actual state of infrastructure and desired future functionality;

* I would like to thank my graduate student, Grzegorz Bera, who undertook deep research of the above mentioned FDL in his master dissertation [11]. His hard work proved that it is not only my general idea, but it really works and is comparable with commercial systems.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 135–142, 2009.

2. preparing project of future network that meets requirements set in first step;
3. modelling of projected networks;
4. physical building of the network.

The process can be also extended with fifth step that covers possible measurements of the network after its deployment.

There are many known approaches used when designing wireless networks. Among them are:

– user deployment – users do not have any network related knowledge or experience; it is based on basic installation of the network components without any analysis or planning; it is simple and it could lead to inefficient coverage [1,2];
– grid installation – an area is divided into $K$ triangles or rectangles, where $K$ is number of available access points; access points are installed in the center of triangles (rectangles); there is no need to know anything on propagation models but this approach is often more expensive than first one and less efficient than next one [2];
– coverage optimization – it is more expensive and complex approach that engages propagation analysis and optimization algorithms when searching for the best access points locations; "best" is understood as providing adequate coverage with minimal infrastructure density [2,3];
– site survey – it is based on the analysis of signal strength's measurements conveyed in the future network area.

### 2.1 Propagation Models

Wireless network design based on the software aids that have built-in propagation models is much cheaper and easier than 'site survey' approach which is time consuming and relativly expensive. It could help to predict signal strength level anywhere in the future network environment [4]. Propagation models could be classified as follows [5]:

– Empirical
  • Free Space Loss
  • One Slope Model
  • Linear Attenuation Model
  • COST 231 HATA
– Semi-empirical
  • Multi Wall Model
  • COST WI
– Optical
  • Ray Traycing
  • Ray Launching
  • Dominant Path Prediction
– Theoretical
  • based on the field theory

Propagation models that have been verified [6] and found to be most prospective during current and future research on wireless network design methods are One Slope Model and Multi Wall Model. Both of them use simple Free Space Loss formula for calculating fundamental loss over the distance in free space.

**Free Space Loss.** Free Space Loss (FSL) model is derived from the Friis Equation [7] which describes signal loss over the distance between the transmitter and the receiver:

$$\frac{P_\mathrm{r}}{P_\mathrm{t}} = G_\mathrm{t} G_\mathrm{r} \left( \frac{\lambda}{4\pi R} \right) \ , \tag{1}$$

where $P_\mathrm{r}$ is power received by the receiving antenna, $P_\mathrm{t}$ is power output to the transmitting antenna, $G_\mathrm{t}$ and $G_\mathrm{r}$ are the antenna gain of the transmitting and receiving antennas, $\lambda$ is the wavelength and $R$ is the distance between the transmitter and the receiver.

Equation (1) transformed and simplified with the assumption that gain of the antennas $G_\mathrm{t} = G_\mathrm{r} = 0\,\mathrm{dBi}$ has the form:

$$PL(R) = 20 \log \left( \frac{4\pi}{\lambda} \right) + 20 \log R \ , \tag{2}$$

where $PL(R)$ is path loss over distance $R$.

After further transformations we get formula:

$$L_\mathrm{FSL}(d) = 32.44 + 20 \log f + 20 \log d \ , \tag{3}$$

where $L_\mathrm{FSL}$ is free space loss [dB], $f$ is the frequency [MHz] and $d$ is the distance [km].

**One Slope Model.** One Slope Model (OSM) is one of the simplest models for calculating of signal strength level without considering structure of the bulding:

$$L_\mathrm{OSM}(d) = L_0 + 10n \log(d) \ . \tag{4}$$

Signal strength level $L_\mathrm{OSM}$ depends mainly on the distance $d$ between the transmitter and the receiver [8]. The enviroment-dependent values $L_0$ (signal loss on 1 meter distance) and $n$ (emprical value representing degree of signal loss) are fixed for specified environment. Values of $L_0$ and $n$ are given in the literature, e.g. for the office environment at frequency $f = 2.45\,\mathrm{GHz}$ variables $L_0$ and $n$ have values: $L_0 = 40.2$ and $n = 4.2$ [8].

OSM is not precise model because it does not incorporate structure of the environment. Sometimes the structure is not known so the model OSM could be useful in such cases.

**Multi Wall Model.** Multi Wall Model (MWM) uses FSL for basic calculation of the loss over the distance and more sophisticated sum of the losses on the obstacles, mainly walls [8]:

$$L_\mathrm{MWM}(d) = L_\mathrm{FSL} + \sum_{i=1}^{N} k_{\mathrm{w}_i} L_{\mathrm{w}_i} + k_\mathrm{f} L_\mathrm{f} \ , \tag{5}$$

where $L_{\mathrm{FSL}}$ is signal loss in free space (Eq. 3), $d$ is the distance [m], $N$ is number of types of the walls, $k_{\mathrm{w}_i}$ is number of the $i$-type walls with $L_{\mathrm{w}_i}$ loss [dB], $k_{\mathrm{f}}$ is number of the ceilings with $L_{\mathrm{f}}$ loss [dB]. Similar to OSM we need some measured values of variables ($L_{\mathrm{w}_i}$ and $L_{\mathrm{f}}$) that are characteristic for modelled environment.

## 3 Floor Description Language

FDL is an aid used in the process of planning of the wireless network. It simply describes environment that is provided to the designer on the map of the floor. Along with the map it is necessary to provide the values of the loss of every type of the walls and ceilings. FDL does not use any propagation model, however one of them is necessary in futher step to find appropriate network components locations. MWM model has been chosen because it seems to be suitable and most reliable in typical applications of home and office designs [6].

### 3.1 FDL Specification

The FDL itself consists of a few markup-like statements that describes the environment as it appears on the floor's map. So the only input data for FDL is map of the floor and specification of loss coefficients for the walls and ceilings.

Types of the walls are represented by different colors: type 1 – red, type 2 – green, type 3 – blue, type 4 – black, type 5 (no wall) – white.

FDL desribes a floor in the building. The structure of the floor is divided into three parallel parts, as shown on Fig. 1. Every part contains at least one room. Parts are declared as follows:

```
number_rooms_A = 5;
number_rooms_B = 3;
number_rooms_C = 4;
```

Rooms could be of different types and adequate identifiers are responsible for their description: stairs (1), corridor (2), office (3), conference room (4), social room (5), toilet (6). For part A on the Fig. 1 we have:

```
id_rooms_A = [1:number_rooms_A; 3 4 3 3 6];
```

Dimensions of the room are declared in next row. First lengths of the rooms are specified and then widths of them.

```
id_dim_A = [1:number_rooms_A; 30 7 8 10 5; 10 10 10 10 10];
```

Key element of the FDL describes all of the rooms' walls. It is done with declarations:

```
walls_A = [1 1 3 2; 3 1 3 2; 3 1 3 2; 3 1 3 2; 3 1 1 2];
```

For all of the specified types of the walls we need to specify losses of them:

```
loss_dB = [15 12 10 40 0];
```

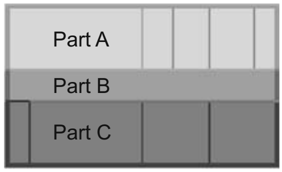



**Fig. 1.** Floor's map

Additional element used only at visualisation stage is thickness of the walls:

```
walls_th = [0.25 0.15 0.1 0.13 0];
```

### 3.2 Modelling Procedures Parameters

FDL needs also some more additional parameters for the modelling procedures. Input data for them are:

– signal level required at the workstation locations [dBm],
– percentage of the coverage [%],
– output power of the transmitters [dBm].

Output data are:

– the best combination of the access points localizations that met all of the requirements.

Propagation procedures are used in the following way:

– signal strength level is computed using MWM formula for every workstation and combination of the access points,
– percentage of the workstations that achieve required level of signal strength is then computed,
– loop is repeated for next combination of the access points, for all of their possible localization,
– the best solution is then chosen – the best means that all of the criteria has been met and number of the access points is the lowest.

Output has also visualisation of the best combination of the access points that assure required parameters.

## 4 Numerical Results

The computations were undertaken in MATLAB environment using aforementioned FDL implementation for sample map of the floor (Fig. 1). Results were compared with output from commercial products – Ekahau Site Survey [9] and RF3D WifiPlanner [10].

Further assumptions and parameters were:

– constant width and length of the area (floor's dimensions: 60 x 25 m),

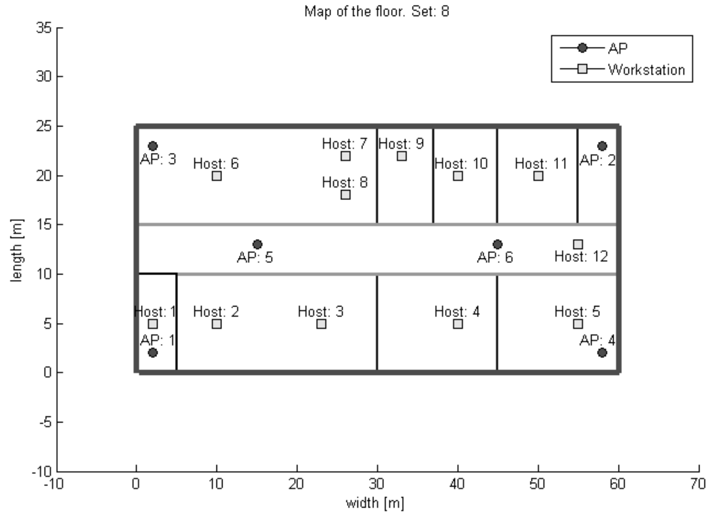



**Fig. 2.** Map of the floor with workstations' and the access points' localizations

**Table 1.** Loss coefficients [dB] and the colors used by the FDL and commercial systems

| Wall type | FDL | Ekahau | RF3D |
|---|---|---|---|
| type 1 | 15/red | 15/orange | 15/red |
| type 2 | 12/green | 15/gray | 12/green |
| type 3 | 12/blue | 15/gray | 10/blue |
| type 4 | 40/black | 45/black | 30/black |

**Table 2.** Coverage percentage [%] results

| Set | No. of AP | FDL | RF3D | Ekahau |
|---|---|---|---|---|
| No. 1 | 1 | 50 | 42 | 42 |
| No. 2 | 1 | 25 | 0 | 17 |
| No. 3 | 1 | 25 | 17 | 17 |
| No. 4 | 2 | 75 | 50 | 67 |
| No. 5 | 2 | 25 | 17 | 17 |
| No. 6 | 2 | 58 | 50 | 58 |
| No. 7 | 4 | 75 | 67 | 75 |
| No. 8 | 6 | 100 | 83 | 92 |

– nine rooms and a corridor,
– four types of the walls (as specified before),
– constant number of workstations and power of the transmitters (15 dBm),
– required percentage of coverage (70%),
– required signal strength level (−63 dBm).

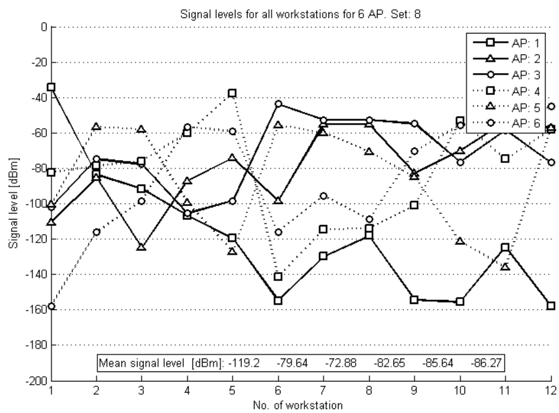



**Fig. 3.** Signal levels' comparison for selected access points combinations

The differences between colors and the loss coefficients used by FDL and commercial systems are summarized in the Table 1.

Sample results given by the FDL implemenation with MWM propagation model are presented on the Fig. 2 and Fig. 3 (map of the floor and signal strength levels comparison respectively). Output graphic results given by commercial systems are omitted.

Numerical results are summarized in the Table 2.

### 4.1 Results Conclusions

Data presented in the Table 2 prove that FDL with MWM can compete with commercial systems when optimization criteria is coverage range. In every case FDL seems to find set of the access points that provides the highest coverage percentage.

Detailed analysis of output data for all of the sets given by three systems (that are omitted due to their volume) lead to following conclusions:

- differences of computed signal levels are not greater than a few dB,
- at least 80% of cases where FDL with MWM is used seems to be better than commercial systems in terms of both coverage range and number of the access points used.

Another research proved that results given by the FDL with MWM are very close to those given by site survey approach. The maximal difference between measurements and FDL with MWM was 4 dB and results given by Ekahau Site Survey were worse in most cases with maximal difference of 9 dB.

## 5 Summary

Presented Floor Description Language although it covers simple idea gives results very close to the results of commercial-grade modelling systems. Its simple ideas that combine geometric decomposition of the floor with classic propagation models could be further developed. Possible enhancements could be done with:

– database extension (e.g. obstacles' definitions etc.),
– auto-localization of the access points procedure,
– additional network parameters,
– graphic files import and recognition of the floor's map.

## References

1. Prommak, C., Kabara, J., Tipper, D., Charnsripinyo, C.: Next generation wireless LAN system design. In: Proceedings of MILCOM 2002, October 7-10, 2002, vol. 1, pp. 473–477 (2002)
2. Unbehaun, M., Kamenetsky, M.: On the Deployment of Picocellular Wireless Infrastructure. IEEE Wireless Communications 10(6), 70–80 (2003)
3. He, J., et al.: Globally Optimal Transmitter Placement for Indoor Wireless Communication Systems. IEEE Transactions on Wireless Communications 3(6), 1906–1911 (2004)
4. Oliver, K.E.: Introduction to Automatic Design of Wireless Networks. ACM Crossroads 11(4) (2005)
5. Luntovskyy, A., Gütter, D., Schill, A.: Models and Methods for WLAN/WiMAX-Network Design. In: Proceedings of 16th International Crimean Conference on Microwave & Telecommunication Technology, vol. 1, pp. 391–393 (2006)
6. Olejnik, R.: WLAN design methods vs. parametric design method (in Polish). In: Wegrzyn, S. (ed.) Sieci komputerowe. Nowe technologie. Wydawnictwa Komunikacji i Lacznosci, Warszawa, vol. 1, pp. 371–380 (2007)
7. Golio, M. (ed.): The RF and Microwave Handbook. CRC Press, Boca Raton (2001)
8. Zvanovec, S., Pechac, P., Klepal, M.: Wireless LAN Networks Design: Site Survey or Propagation Modeling? Radioengineering 12(4), 42–49 (2003)
9. Ekahau Site Survey, http://www.ekahau.com
10. RF3D WifiPlanner, http://www.psiber.com/rf3d/rf3d.html
11. Bera, G.: An algorithm of WLAN elements' localization based on the floor's map (in Polish). Master thesis, Szczecin University of Technology, Szczecin (2008)

# Dependencies and Configurations of Solutions in Multicriteria Optimization in Nets

Henryk Piech

Czestochowa University of Technology, Dabrowskiego 73
h.piech@adm.pcz.czest.pl

**Abstract.** We want, in general form, to close to the practical aspects of multicriteria optimization in nets. Set of criterions contains typical aims for network task such critical path, minimal path, minimum spanning tree, maximal flow, cardinality matching etc. Once criteria can be multiplied by, on example, several starting and target points. The others by mutually covered sets of constrains and dependences between them. In practice also we have deal with uncertain parameters characterizing elements of network structure. In such case we will choose aggregation criteria variant and the best with point of view of given criterions set of values of parameters. Practical validity of such kind of analyses consist in utility obtained results in projects or their modification during exploitation.

## 1 Introduction

Many problems concerning network creation are caused by the necessity to account few criteria simultaneously. These problems apply to creation and operation of roads [13,17,1], undergrounds [21,23,12], pipelines, energy lines [22,5], and also computer networks. The methods used depend on given situations, with which they are connected. For instance: some specific constraints and simplifications are used, they concern also (here we list them) creation of net substructures [9], use of specific techniques and routing conventions [11,4,2], and also resulting from topological reservations [6]. It is often essential to take into consideration natural conditions (geographical, geological etc.) [8,12], multimodality of urban structures [16,3,14]. The variety of criteria sets is very broad. These criteria are connections of such goals as the shortest ways, maximum safety, minimal costs, minimal problems with realisations, maximum flow, optimal organisation, most numerous associations, minimum monochromatic number etc. From the point of view of algorithmic and mathematic optimalisation aspects many different attitudes are used, some of them are: approximated [7,10], regressive and combinatorial techniques [15] etc. The compromise for a set of criteria can be received in many ways and often depends on the form of criteria or attributes' aggregation [24].

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 143–150, 2009.

## 2 Aggregation of Criterions of the Minimal Path in Reference to Different Target Points

The problem of extending net structure is often connected with multiplying start or target points i.e. with repeating task of shortest path. Sometimes it is possible to realize these tasks with help of the minimum spanning tree algorithms but sometimes with help of using several times criterions of the shortest path [18] (Fig. 1).

Utilization more rigorous criterion (based on minimum spanning tree method) to realization the task (which does not require finding minimal paths to all nodes) is obviously less profitable. The adding node 7 requires the correction of connections considered and defined at the projecting and exploitation stages. There are possible ways of connecting knot 7: 1-7, 2-7, 3-7, 4-7, 5-7, 6-7. Choosing two criterions of the shortest paths (to given target nodes 4 and 7) we already on projecting stage take into account the sequential stage process of optimization of connections basing on two criterions. In this way we receive more effective solution. We can describe it as follows (see Fig. 2):

$$\begin{aligned} 1-2-4-3-6-7 \text{ (minimal spanning tree)} > \\ 1-2-4+1-5-7 \text{ (2 shortest independed path)} > \\ 1-2-4-7 \text{ (2 shortest depended path – staged)} \end{aligned} \tag{1}$$

The synthesis of task considered described situation can lead to solution the next problem:

*Problem 1.* The location of seven agencies is described on Fig. 2. In every agency the accessories (components) are produced. The components of modules are assembled only in agencies 4 and 7. Modules are transported on appointed roads among main agency 1 and agencies 4 and 7. The task consist in designing the structure of inferior roads in such way that both criteria i.e. building and exploitation (the transportation) will be the cheapest.

*Choice and adaptation criteria function.* The transportation of accessories between nodes suggests use the minimum spanning tree criterion. Transport between

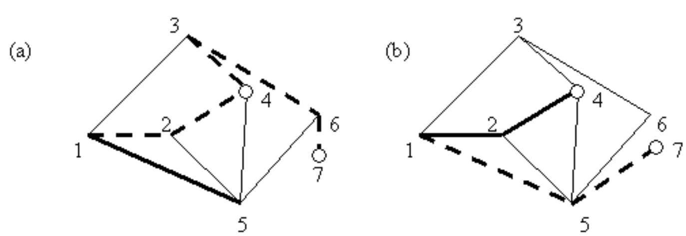



**Fig. 1.** The difference effects of solution of problem of multiple minimal paths. Utilization minimum spanning tree methods (a) and double using the shortest paths method: to knots 4 and 7 (b)

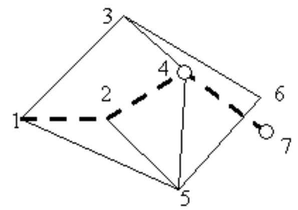



**Fig. 2.** The most profitable variant ways of reaching to given nodes 4 and 7

knots 1-4 as well as 1-7 can be optimized by two dependent criterions of minimum paths (considered sequentially with regard previously got results): Fig. 2.

## 3 The Aggregation of Criteria of Maximum Flow and Minimum Spanning Tree

The building and extension of energetic nets transferring the varied medias is connected with task requiring the several criterions of optimization. The set of criterions is the reflection the character of transportation different medias too.

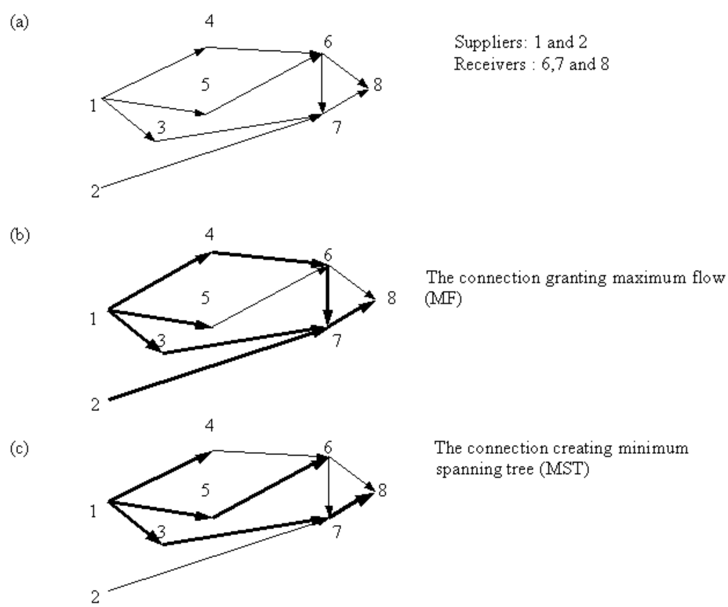



**Fig. 3.** The independent investigation the criterions of maximum flow and minimum spanning tree

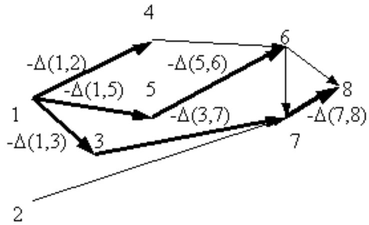



**Fig. 4.** The correction of flow capacity in connections creating MST

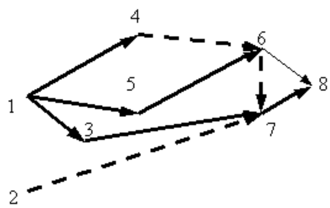



**Fig. 5.** Effect of aggregation of criterions MST and MF: the complement to sum of sets was introduced with help of dashed line

In concrete situations we note down the necessity of regard of flow capacities, and also the length of connections. In most situation we have deal with directed graphs. The cost of realization of connections depends on length and, for example, on diameters of link. The example – scheme of network structure is illustrated in Fig. 3.

The solution based on criterion of maximum flow will be essential in net exploitation . The solution obtained on base of minimum spanning tree criterion, that means the skeleton of connections, playing the decisive role on stages of building (or extension) the net. It is the main factor formatives the costs of network structure creation. The synthesis of above described problem can have following form:

*Problem 2.* Drawing energetic medias from sources 1 and 2 we have to satisfy the demand of factories 6, 7 and 8. It would to be as big as possible of medium amount. The flow capacity of connections be defined. by technical and security features. The private recipient make up of the secondary importance (in relation to size of orders) group of supplied of medias . They are distributed in nodes 4, 5, 6, 7, 8, 9. Factories have the possibility of storing medias, and private recipient obviously not.

*Choice and adaptation criteria function.* For description the supplies of factories we use analysis leaning on method of maximum flow criterion multiplied by two sources ( start points) and three target locations. For description of private recipient's supply we use minimum spanning tree algorithms sometimes extended by the cheapest flow method. Technical regards suggest that the first stage i.e. project will be connected with private recipient's supply. It will correct flow capacity of constant values (answering to private recipient supplies) in creating minimum spanning tree connections (Fig. 4).

The aggregation of criteria influences is realized as the sum of set of active elements realizing the supply of factories and private recipient (Fig. 5).

## 4 Aggregation of Criterions of the Cardinality Matching and the Shortest Path

The wireless connection requires the association sender and recipient every time. Such arrangement in set of nodes give chance to obtain maximum number of associating equals $\lfloor count\_w/2 \rfloor$, where $count\_w$ – the number of nodes. The way of matching depend on number of free nodes. The occupation nodes can depend on unfinished form of exploitation in previous associations, or on different forms of nodes engagement, or on date transfer with preference meaning (Fig. 6).

To distributed conversion is used the transfer between nodes 1-2-6 (bold continuous line). Remaining nodes 3, 4, 5, 7, 8, 9 can realize the communication connections, within number of the cardinality matching in different possible configurations. Obviously different scales of net occupations will determine the number and configuration, free and designed to connections of communication, nodes. Exploitation of such net will require continuous solutions of optimum elements configurations from its structure. If net structure functioning based on wireless connections the possible contacts can consist in exploitation the whole graph. However, in the criterion of the cardinality matching problem decisive role play "free" nodes. The synthesis of task of multicriteria optimization can have following figure:

*Problem 3.* The conversion of preference data requires engaging following nodes (for example station sending-receiving): 1, 2, 6. Remaining nodes can be used to realization the communication connections. Every of nodes can play role of sender's or recipient's function in given moment for concrete individual connection.

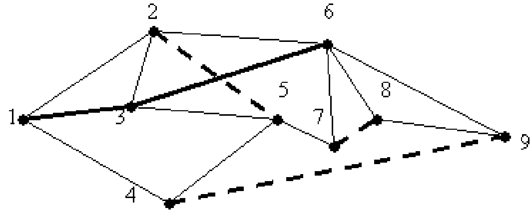



**Fig. 6.** Location of associating after exclusion the nodes used in distributed conversion

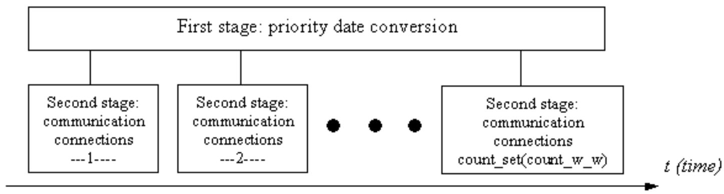



**Fig. 7.** Location of associating after exclusion the nodes used in distributed conversion

*Choice and adaptation criteria function.* In first stage for preparation of date conversion we will use the criterion of the shortest path. In dependence from kind of conversion (it can be minimum spanning tree too, optimal scheduling or minimum critical path, too). Second stage consist in finding the largest cardinality matching for the shortest communication connections. Several sets of the largest cardinality matching during date conversion can be realized simultaneous with excluding the previous sets connections. That means that it appears the scheduling problem of matching. Generalizing, the number of compositions can be defined: $count\_set(count\_w\_w) = c * (c-2) * (c-4) * \ldots$, where $c = \lceil count\_w\_w/2 \rceil * 2 - 1,\ c > 0,\ count\_w\_w$ – the number of free nodes. The full configuration of exploitation of nodes is illustrated in Fig. 7.

## 5 Aggregation of Criteria of the Cheapest Flow and Scheduling Optimization

Generally, we would choose the shortest path among possible critical paths. In deter-ministic tasks critical path we can choose every critical path with different configuration under condition, that their lengths will be the same as the shortest. In conditions of uncertain knowledge about nets parameters we get the different variants the critical paths among which we will choose the set of connections with the shortest length which fulfill constrains or added criteria (e.g. connected with probability, membership function etc.). Solution which we associate with optimum scheduling is a structure of critical path for given weight parameters. In respect on fuzzy parameters character we can choose such set of them, which will answer the most profitable criteria characteristics. We assume additionally, that exists the set of constrains which have to be fulfilled, and which involve some parameters simultaneously. The task synthesis of multicriteria optimization can suggest next form:

*Problem 4.* The times of realization of investment stages can oscillate in given ranges. They are additionally connected by set of constrains introduced in classic form that means system of inequalities [18]. Task consists in such establish temporary values of parameters responding individual stages to warrant the optimum realization of investment.

*Choice and adaptation criteria function.* The connected with optimization of process criterion of scheduling we look for systematically or randomly configured sets of parameters in given interval. Set of constrains which have to be fulfilled in connection with finding the shortest critical path that is the second stage of optimization.

## 6 Conclusions

Optimization in nets usually has muticriterial character, what infers not only from necessity of matching several aims but also from multiplying start and target elements.

Generalization criteria tasks leads to creating the algorithm of finding the optimum solution both under in relation to structure and its parameters. Essential problem refers to algorithm consist in establish the order of taking into account criterions as well as ways of aggregations partial optimal solutions.

Fuzziness of weight parameters leads to necessity of creating the permutation of date arrangement what in substantial way increases complexity of algorithm (in pessimistic variant the complexity may increase itself about $kn$ times, where the $k$ – the number of steps in fuzzy (interval) range, $n$ – the number of connections).

The utilization of Solvers give the possibility of quick location maximum of membership function which in this case can to be definite as function "desirable". Advantage of Solvers utility is possibility of regarding freely numbers of constrains.

## References

1. AASHTO: A policy on geometric design of rural highways, Washington (1975)
2. Aboelela, E., Douleligeris, C.: Fuzzy multiobjective routing model in B-ISDN. Computer Comunication 21, 1571–1584 (1998)
3. Arnold, P., Peeters, D., Thomas, J.: Modeling a rail/road intermodal transportation system. Transportation Research Part E 40, 255–270 (2004)
4. Ash, G.R.: Design and Control of networks with dynamic nonhierarchical routing. IEEE Comm. Mag. 28(10), 34–40 (1990)
5. Bella, A., Ducstein, L., Sziradovski, F.: A multicriterion Analysis of the water Allocation. Conflict in the Upper Rio Grande Basin. Applied Mathematics and Computation, New York (1966)
6. Bertault, F.: A force – directed algorithm that preserves edge crossing properties. In: Kratochvíl, J. (ed.) GD 1999. LNCS, vol. 1731, pp. 351–358. Springer, Heidelberg (1999)
7. Blum, A., Ravi, S., Vempala, A.: A constant factor approximation for the k-st problem. In: Proc. 28th Ann. ACM Symp. on the theory of Computing (StOC 1996), pp. 442–448. ACM Press, New York (1996)
8. Bookbiuder, J., Fox, N.: Intermodal Routing of Canada Mexico shipments under NAFTA. Transportation Research E,34, 289–303 (1998)

9. Drangmeister, K.U., Krumke, S.O., Marathe, M.V., Noltemeier, H., Ravi, S.S.: Modifying edges of a network to obtain short sub graphs. Theoretical Computer Science 203, 91–121 (1998)
10. Goemas, M.X., Golberg, A.V., Plotkin, S., Shmoys, D.B., Trados, E., Wiliamson, D.P.: Improved approximation algorithms for network design problems. In: Proc. 5th Ann. ACM-SIAM Symp. on Discrete Algoritms (SODA 1994). ACM Press, New York (1994)
11. Hollendorn, W.A.H., Seising, R., Weitzel, C.T.A.: Fuzzy Routing. Fuzzy Sets and Systems 85, 131–153 (1997)
12. Hong, S.-H., Merrick, D., Nascimento, H.A.D.: The metro map layout problem. In: Pach, J. (ed.) GD 2004. LNCS, vol. 3383, pp. 482–491. Springer, Heidelberg (2005)
13. Karamaras, G.S., Brino, L., Corrieri, G., Pline, C., Grasso, P.: Application of Multicriteria Analysis to Select the Best Highway Alignment. Tunneling and Underground Space Technology 15(4), 415–420 (2000)
14. Kozan, E.: Optimizing container transfers at multimodal terminals. Mathematical and Computer Modeling 31, 235–243 (2000)
15. Lawler, E.L.: Combinatorial Optimization. In: Networks and Matroids, ch. 4. Holt Rinhart and Winston, New York (1976)
16. Modesti, P., Sciomachen, A.: A utility measure for finding multiojective shortest paths in urban multimodal transportation networks. European J. of Oper. Res. 111, 495–508 (1998)
17. Perez de la Cruz, J.L., Conejo-Munoz, R., Morales Bueno, R.: Highway design by constraint specification. Artificial Intelligence in Engineering 9, 127–139 (1995)
18. Piech, H., Ptak, A., Machura, M.: Generation of multicriterial optimization solution on the basis of uncertain evaluation. In: 18th International Conference on Multiple Criteria Decision Malking, Chania (2006)
19. Piech, H., Ptak, A., Machura, M.: Fuzzy Strategies in Modelling Effect of Technological Inplementation. In: 2nd Inernational Conference on Fuzzy Sets and Soft Computing FSSCEF, Saint Petersburg (2006)
20. Rai, S.: A cut set approach to reliability in communication networks. IEEE Trans. Reliability R-31, 428–431 (1982)
21. Roy, B., Present, M., Silhol, D.: A programming method for determining which Paris metro station should be renovated. European Journal Operational Research 24, 318–334 (1986)
22. Roy, B., Slowiński, R., Treichel, W.: Multicriteria programming of water supply systems for rural areas. Water Resour. Bull. 28(1), 13–32 (1992)
23. Stott, J.M., Rodgers, P.: Metro Map Layout Using Multicriteria Optimization (2001)
24. Yager, R.R.: On ordered weighted averaging aggregation operators in multicriteria decision making. IEEE Transaction on Systems, Man and Cybernetics 18(1) (1998)

# Web Traffic Modeling for E-Commerce Web Server System*

Leszek Borzemski[1] and Grażyna Suchacka[2]

[1] Institute of Computer Science,
Wrocław University of Technology, Wrocław, Poland
leszek.borzemski@pwr.wroc.pl
[2] Chair of Computer Science,
Technical University of Opole, Opole, Poland
g.suchacka@po.opole.pl

**Abstract.** The paper concerns a problem of the e-commerce Web server system performance evaluation through simulation experiments, especially a problem of modeling a representative stream of user requests at the input of such system. Motivated by a need of a benchmarking tool for the Business-to-Consumer (B2C) environment we discuss a workload model typical of such Web sites and also a model of a multi-tiered e-commerce Web server system. A simulation tool in which the proposed models have been implemented is briefly talked over and some experimental results on the Web system performance in terms of traditional and business performance measures are presented.

## 1 Introduction

During the last decade one could observe a huge increase of interest into a field of Quality of Web Service (QoWS). In response to a very significant problem of degradation and unreliability of the Web service a lot of research in that field has been done. A great deal of them characterizes the workload at the Web server input and proposes various mechanisms aimed at improving the Web server performance under overload. The majority of these research concern Web servers providing information, entertainment or multimedia services. Relatively slow attention has been spent on commercial Web servers so far. Data on e-commerce traffic recorded in Web server logs are hardly available because of a sensitive financial aspect of companies' revenue. There is no suitable benchmarking tool for the e-commerce Web server performance evaluation through simulation experiments.

Popular benchmarks such as httperf, SURGE, WebBench or WebStone are not suitable for the e-commerce Web server systems due to a simplified workload model. Only TPC-W benchmark specification defines a workload model oriented to e-commerce transactions [1]. However, it doesn't model details of the

* This work was supported by the Polish Ministry of Science and Higher Education under Grant No. N516 032 31/3359 (2006–2009).

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 151–159, 2009.

Web server resources usage at the HTTP level which is needed for the system performance evaluation in the laboratory environment. Available TPC-W implementations don't provide business-oriented metrics and don't model various user profiles.

For the above reasons we decided to work out a workload model typical of e-commerce Web sites based on the up-to-date literature and to develop a simulation tool providing the following functionalities:

- generating many concurrent user sessions at a given session arrival rate and providing session-oriented statistics;
- modeling two session classes: KC class for key customers of the store (who can be identified after logging into the site and are characterized by a greater probability of making a purchase) and OC class for ordinary customers;
- modeling interaction between users and the Web site, especially the impact of the Web system performance on the user behavior;
- generating very variable and self-similar Web traffic typical of e-commerce sites at the HTTP level for a mix of static, dynamic and secure requests with different Web system resource consumption profiles;
- providing not only traditional Web server performance measures but also business-oriented measures connected with the revenue.

Section 2 discusses session-based workload model used in our workload generator and Sect. 3 describes a model of multi-tiered e-commerce Web system implemented in the simulator. In Sect. 4 architecture of our simulation tool is briefly presented and some experimental results are presented in Sect. 5. We summarize in Sect. 6.

## 2 Session-Based Workload Model

An e-commerce workload is composed of many user sessions. A user session is defined as a sequence of logically and temporally related requests issued by the user during a single visit to the site. During the session the user performs some typical business operations such as browsing and searching goods, adding them to a shopping cart, etc. Considering user behavior at the Web site we have to differentiate between two session classes: KC class for key customers and OC class for ordinary customers. Key customers have different navigation patterns at the site and are characterized by a greater probability of making a purchase. Our user session model at the e-commerce Web site is based on a Customer Behavior Model Graph (CBMG) proposed in [2]. We model sessions of OC class and KC class according to modified CBMGs for an occasional buyer profile and a heavy buyer profile, respectively.

In original CBMG six session states were distinguished: H (Home), B (Browse), S (Search), D (Details), A (Add) and P (Pay). However, interactions involved in user identification at the site such as user registration and logging on weren't taken into consideration so we augmented the CBMG with two additional states corresponding to Web interactions R (Register) and L (Login). A rationale for such

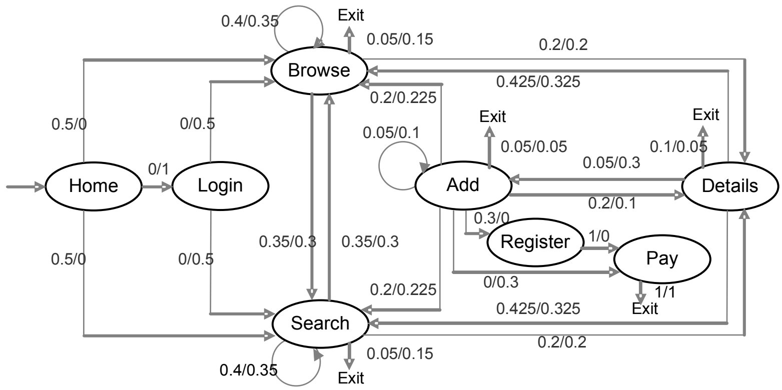



**Fig. 1.** Modified CBMG used to model ordinary customer session/key customer session

approach is a necessity to include an additional computational overhead incurred in performing login and register Web interactions in simulation experiments and thus making service times for both session classes more realistic.

CBMG used in our workload model is presented at Fig. 1. Nodes of the graph correspond to the session states. Arrows between states mean transition probabilities between the states for both profiles whereas arrows *Exit* mean the user spontaneous decisions on leaving the site. Each transition is characterized by a mean value of user think time which is modeled according to an exponential distribution with a maximum of 10 times the mean [1]. Mean user think time $t_{k,l}$ is 15 sec for all transitions except the following ones: $t_{\mathrm{S,D}} = 30\,\mathrm{sec}$, $t_{\mathrm{D,A}} = 45\,\mathrm{sec}$ and $t_{\mathrm{A,P}} = 25\,\mathrm{sec}$ [2].

When a new session is initiated, a session class KC or OC is assigned to it according to pre-specified probability values $p_{\mathrm{KC}} = 0.1$ and $p_{\mathrm{OC}} = 0.9$, respectively. These values were chosen as a result of preliminary simulation experiments so as to provide a realistic percentage of all generated purchase sessions i.e. about 5% [3].

Initial state of each session is H. Each next Web interaction within the session is computed based on the transition probabilities from the current state to other states. However, it depends also on the response time of the last Web interaction: if the Web system performance is poor and the page response time is longer than timeout value $T_{\mathrm{u}}$, the user grows impatient and leaves the site and consequently his/her session is aborted. We set $T_{\mathrm{u}} = 8$ seconds according to the well known 8-second rule.

Since we use a business-oriented system performance measures it is essential to incorporate a financial aspect of users' activities at the site into the workload model. We simulate an electronic bookstore and prices of books added by customers to the carts are generated according to a truncated Gaussian distribution (model parameters are presented in Table 1) [2]. In our model the number of books being purchased in a single transaction depends on the number of visits to the state A during the session.

**Table 1.** Workload model parameters determining product prices

| **Category** | **Technical books** | **Non-technical books** |
|---|---|---|
| Percentage of books selected | 20% | 80% |
| Average price | \$45.00 | \$18.00 |
| Price range | [\$5.00, \$100.00] | [\$5.00, \$60.00] |

**Table 2.** Distributions and their parameters applied in a workload model

| **Category** | **Distributions** | **Parameters** |
|---|---|---|
| Number of static objects per Web page | Pareto | $\alpha = 1.33$, $k = 2$ |
| Number of dynamic objects per page | Geometric (+ 1) | $p = 0.8$ |
| HTML object size | Body: Lognormal | $\mu = 7.63$, $\sigma = 1.001$ |
| | Tail: Pareto | $\alpha = 1$, $k = 10240$ |
| Embedded static object size | Lognormal | $\mu = 8.215$, $\sigma = 1.46$ |
| Embedded dynamic object size | Weibull | $\lambda = 0.0059$, $b = 0.9$ |
| Interarrival time of hit requests | Weibull | $\alpha = 7.64$, $k = 1.705$ |

Each Web interaction corresponds to a single Web page request. We model all Web interactions as dynamic pages with many static objects and maximum ten dynamic objects. Furthermore, Web interaction P is modeled as the SSL-secured transaction. Executing each Web page request requires processing many HTTP requests, i.e. a hit for an HTML page and following hits for all objects embedded in it. According to our best knowledge there is no workload model encompassing all aspects of the e-commerce workload at the HTTP level so we decided to combine in our model results of several workload studies. We believe this combination is well justified given the common base of the studies i.e. the Web server.

The number of static objects per page is modeled by a Pareto distribution and the number of dynamic objects is obtained by a geometric distribution and incremented by 1. Sizes of HTML files are obtained from a hybrid function where a body follows a lognormal distribution while a tail follows a Pareto distribution. Sizes of embedded static and dynamic objects are obtained from a lognormal and Weibull distribution, respectively. Interarrival time of hit requests is given by a Weibull distribution. Table 2 specifies workload distributions along with their parameters [4,5,6,7,8].

## 3 E-Commerce Web System Model

An e-commerce site is typically organized as a multi-tier application in which request processing is realized at three logical software layers: Web server, application server and database server layer. We decided to model a three-tiered e-commerce system based on the approaches proposed in [5,9] and analyzing results of business workload characterization studies [10,11].

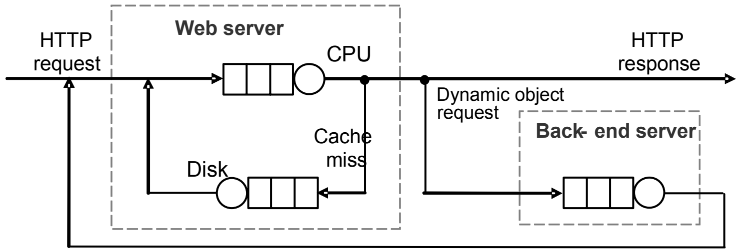



**Fig. 2.** Queuing network model of the e-commerce Web system

System model includes two main components: a Web server and a back-end server (i.e. an application server and a database server). Since our research concentrates on the performance of the Web server system itself we do not model network components engaged in request processing and transfer. We do not consider a cluster-based Web system either, wanting to focus just on key resources being potential bottlenecks of a typical e-commerce Web system in B2C environment.

System model is based on queuing networks [12]. A Web system is depicted as a network of service centers among with queues to them in which client requests wait for a service (Fig. 2). Requests enter the system from the outside and leave it after the service is completed. Applying a queuing model allows to capture crucial relationships between key Web system parameters and to compute performance measures as a function of the system load.

A Web server model contains two service centers corresponding to key server resources: CPU and disk. Every admitted HTTP request first enters the CPU queue. CPU parses the request and decides about the way of handling it. For a static request a file is brought from the cache or from the disk. Dynamic requests go to the back-end server queue. After completing a response file the CPU sends it back to the client. We model a cost of HTTP request processing at the Web server after [9] where costs for static request processing steps were determined. Taking into consideration much higher speed of contemporary processors and disks, we divided the component costs measured in [9] by 10 [13]. An overall cost of processing a single static HTTP request at the Web server includes a cost of CPU processing (i.e. a cost of connection establishment and teardown $145/10\,\mu$s each and a cost of transmit processing $40/10\,\mu$s per each 512 B) and in case of a cache miss a cost of serving a file at a disk; it includes a cost of reading a file from disk (28/10 ms), the file transfer time ($410/10\,\mu$s per each 4 KB) and additional time for files larger than 44 KB (14/10 ms per each 44 KB). A probability of cache hit for a static request is 88%.

Back-end server model simulates activities involved in dynamic HTTP request processing in the application and database tier. A back-end node is modeled as a single service center with a single queue like in [5]. A dynamic request processing cost includes an additional overhead incurred by generating a dynamic content through invoking the database.

In e-commerce systems dynamic request execution times vary significantly depending on the kind of a business operation [10]. We model these times according to an exponential distribution with the average depending on a session state. We propose applying mean time values resulting from TPC-W servlet execution costs measured in [10]. We divided eight Web interactions into three categories, each one with different mean request service time in a back-end server: a category *Very-Intensive* for the Web interaction S, a category *Intensive* for the Web interaction P and a category *Light-intensive* for interactions H, B, D, A, L and R. Average service times for these categories amount to 91 ms, 54.26 ms and 0.56 ms, respectively.

Web interactions connected with finalizing a transaction at the B2C sites are typically realized over secure connections using SSL or TSL protocol. In e-commerce system it causes an average increase of system response times of about 5% and it can be attributed mainly to the application server CPU [11]. Therefore our model includes an additional overhead for Web interaction P requests, realized by increasing back-end node service times obtained for requests in a state P by 5%.

## 4 Simulation Tool

Based on the workload model described above we developed a synthetic workload generator as an integral module of the simulator presented in Fig. 3. A simulation tool was implemented in C++ with CSIM19 [14], a toolkit for modeling complex systems.

Session-based workload is generated in real time as it depends on the Web system efficiency. In QoS module we implemented a variety of classification, admission control and scheduling algorithms which are applied based on input parameters and based on the current system load level reported by the workload monitor module. The output module collects simulation statistics and produces final reports.

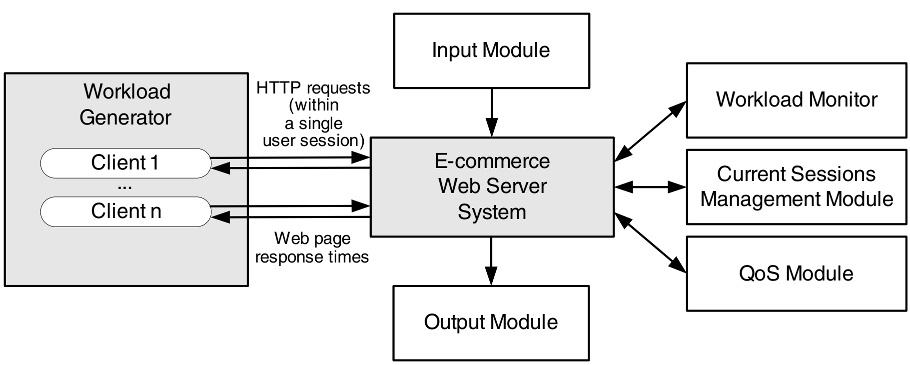



**Fig. 3.** Architecture of the e-commerce system simulation tool

## 5 Simulation Results

Due to space limitations only some results for FIFO scheduling in the system queues are presented. Every single experiment was run for a constant session arrival rate. In consecutive experiments the session arrival rate was gradually increased. Simulation results are illustrated in Fig. 4–5.

Figure 4a shows the Web system throughput in the number of successfully completed user sessions as a function of the session arrival rate. The three curves in the figure represent throughputs for all sessions, ordinary customer sessions (OC) and key customer sessions (KC), respectively. For our workload model the Web system reaches its maximum capacity in sessions at about 70 new sessions per minute and after that point the throughput continues to drop rapidly. A sharp slope of the curves is due to long page response times leading to users' impatience and aborted sessions. At a session arrival rate of 80 sessions/min none of the sessions is completed although the system throughput in the number of completed HTTP requests still increases (Fig. 4b).

Figure 5a presents 90-percentile of page response time, mean page response time and median of page response time for all users. Page response times increase gradually until the system nears its maximum capacity in sessions and then goes up rapidly to an unacceptable level, much exceeding 8 seconds.

Figure 5b depicts the revenue throughput in \$ per minute and potential revenue losses per minute (computed as \$ per minute that were accumulated in shopping carts of the sessions that had been aborted due to a poor Web system performance). As the load increases the potential revenue losses increase till the point indicating the maximum system capacity in sessions and above that point start to decrease. It is caused by the fact that sessions are aborted at early stages and the customers haven't had a chance of adding products to their carts so the potential revenue (and their losses) is lower. Revenue rate grows linearly until the server nears its maximum capacity. However, it's worth observing that although the maximum system capacity in sessions is about 70 sessions/min, the maximum revenue throughput is reached at lower load of about 60 sessions/min.

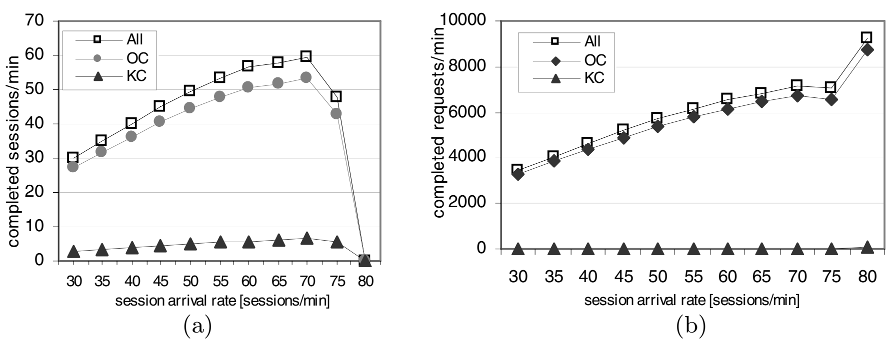



**Fig. 4.** (a) System throughput as the number of user sessions completed per minute; (b) The number of HTTP requests completed per minute

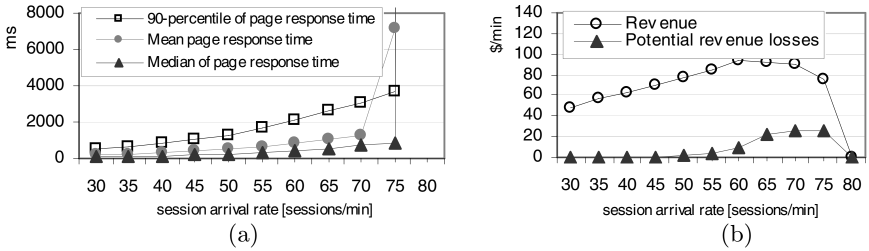



**Fig. 5.** (a) Page response times; (b) Revenue and potential revenue losses per minute

## 6 Concluding Remarks

Similarly to the results reported in the literature on the quality of Web service, our simulation results clearly demonstrate that the overloaded Web server system experiences severe service degradation in terms of traditional performance measures such as throughput and page response times and also in terms of business measures such as revenue gained through the successfully completed user transactions. Such observations motivated us to design new admission control and scheduling algorithms aiming at optimizing the business-oriented Web system performance measures. Our future works concern verifying their efficiency using our simulation tool.

## References

1. García, D.F., García, J.: TPC-W E-Commerce Benchmark Evaluation. Computer 36(2), 42–48 (2003)
2. Menascé, D.A., Almeida, V.A.F., Fonseca, R., Mendes, M.A.: Business-Oriented Resource Management Policies for E-Commerce Servers. Perf. Evaluation 42(2–3), 223–239 (2000)
3. Nielsen, J.: Why people shop on the Web (February 1999) (April 2002), http://www.useit.com/alertbox/990207.html
4. Barford, P., Bestavros, A., Bradley, A., Crovella, M.: Changes in Web Client Access Patterns: Characteristics and Caching Implications. WWW 2(1–2), 15–28 (1999)
5. Cardellini, V., Casalicchio, E., Colajanni, M., Mambelli, M.: Web Switch Support for Differentiated Services. ACM Performance Evaluation Review 29(2), 14–19 (2001)
6. Casalicchio, E., Colajanni, M.: A Client-aware Dispatching Algorithm for Web Clusters Providing Multiple Services. In: 10th International WWW Conference, pp. 535–544 (2001)
7. Shi, W., Collins, E., Karamcheti, V.: Modeling Object Characteristics of Dynamic Web Content. Journal of Parallel and Distributed Computing 63(10), 963–980 (2003)
8. Xia, C.H., Liu, Z., Squillante, M.S., et al.: Web Traffic Modeling at Finer Time Scales and Performance Implications. Performance Evaluation 61(2–3), 181–201 (2005)

9. Pai, V.S., Aron, M., Banga, et al.: Locality-Aware Request Distribution in Cluster-based Network Servers. ACM SIGOPS Operating Systems Review 32(5), 205–216 (1998)
10. Elnikety, S., Nahum, E., et al.: A Method for Transparent Admission Control and Request Scheduling in E-Commerce Web Sites. In: WWW, pp. 276–286. ACM Press, New York (2004)
11. Paixão, G.T., et al.: Design and Implementation of a Tool for Measuring the Performance of Complex E-Commerce Sites. In: Haverkort, B.R., Bohnenkamp, H.C., Smith, C.U. (eds.) TOOLS 2000. LNCS, vol. 1786, pp. 309–323. Springer, Heidelberg (2000)
12. Czachórski, T.: Modele Kolejkowe w Ocenie Efektywności Pracy Sieci i Systemów Komputerowych. Pracownia Komputerowa J. Skalmierskiego, Gliwice (1999) (in Polish)
13. Borzemski, L., Zatwarnicki, K., Zatwarnicka, A.: Adaptive and Intelligent Request Distribution for Content Delivery Networks. Cyber. and Systems 38(8), 837–857 (2007)
14. Schwetman, H.: CSIM19: A Powerful Tool for Building System Models. In: WSC, pp. 250–255. IEEE, Los Alamitos (2001)

# RULEGO Bioinformatical Internet Service – System Architecture*

Aleksandra Gruca, Marek Sikora, Łukasz Chróst, and Andrzej Polański

Institute of Informatics, Silesian University of Technology, Gliwice
{Aleksandra.Gruca,Marek.Sikora,Lukasz.Chrost,Andrzej.Polanski}@polsl.pl

**Abstract.** In this paper we present the architecture of the RuleGO – a grid-based Internet application for describing gene groups using decision rules based on Gene Ontology terms. Due to the complexity of the rule induction algorithm there is a need to use sophisticated mechanisms for supporting multiple requests from the application users and to perform simultaneous analysis of gene groups in a distributed environment.

## 1 Introduction

DNA microarray chips [2] are one of the most indispensable tools used in biological and medical laboratories all over the world. They allow for simultaneous recording of thousands of gene expression profiles in a single biological experiment. Such experiments may be focused on discovering genes that express differentially between two different groups of donors (usually one of them suffers from particular disease and the other is a control group) or on measuring time courses of genes expression under some experimental conditions.

Very important step of the DNA microarray experiment is a biological interpretation of the interesting groups of genes obtained during the DNA microarray experiment. Selected groups may include genes that have similar expression values or genes that have significantly different expression values in comparison to other genes whose probes are placed on DNA microarray chip. Biological interpretation of the obtained groups is usually done by an expert in the field, frequently by manual reviewing genes composing the groups, which requires lots of experience and is time consuming. However, an expert work may be supported by including in the analysis functional annotation of genes and applying statistical methods or data mining techniques to discover non trivial dependences among genes composing the clusters.

One of the popular and widely used tool for describing genes is Gene Ontology database [1]. The Gene Ontology database is a hierarchical structure that includes a vocabulary describing genes and gene products in the form of three disjoint directed acyclic graphs (DAGs) representing biological process, molecular function and cellular component. The information included in GO database

* This paper was partially supported by the European FP6 grant, GENEPI-lowRT, Genetic Pathways for the Prediction of the Effects of Ionising Radiation: Low dose radiosensitivity and risk to normal tissue after Radiotherapy.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 160–167, 2009.

is represented by GO terms – single units that describe gene and gene products. The hierarchy of DAGs reflects the structure of information included in GO terms – terms that are close to the root describe general concepts, while terms lower in the hierarchy provide more specific description.

Typical analysis that utilizes GO terms for gene groups description involves application of statistical tests in order to detect the terms that are under- or overrepresented in the analyzed gene group. There are many tools dedicated to such type of analysis which are available as stand alone applications or as web-based systems. Description and comparison of the most of the GO processing tools can be found in paper [7]. Apart from searching for a list of single GO terms describing genes, one can be interested if there exists a statically significant combination of GO terms in analyzed gene group. Such combinations of GO terms may be discovered with the use of rule induction algorithms. Based on the decision rules that include GO terms in the conditional part one can obtain a description of gene groups [5].

In order to make the rule induction method publicly available we developed the Internet application RuleGO [11]. Due to the high resource requirements of the method we designed a system that allows performing simultaneous analysis in a distributed environment. In this paper we present a grid-based architecture of that system.

The paper is organized as follows. Section 2 contains a short description of the decision rules induced for description purposes. Section 3 presents architecture of the system. Section 4 shortly describes graphical user interface and includes an example of an analysis performed with the use of RuleGO application. Finally, in Sect. 5 conclusions and future work is included.

## 2 Decision Rules

Let there be a set of $n$ genes $U = \{x_1, x_2, \ldots, x_n\}$ whose probes are placed on DNA microarray chip. Any gene from the set $U$ may be described by GO terms that are related to biological functions of that gene. Treating GO terms as binary attributes that may be used to describe a set of objects (genes) we can form a decision table $DT = (U, A \cup \{d\})$, where $U$ is a set of genes, $A$ is a set of attributes describing these genes and $d$ is a decision attribute that represents a specific number of the gene group. Based of the decision table we can generate decision rules in the following form:

$$\text{IF } a_1 \in V_{a1} \text{ and } a_2 \in V_{a2} \text{ and } \ldots \text{ and } a_n \in V_{an} \text{ THEN } d = v \ , \tag{1}$$

where $v \in D_d$, $\{a_1, a_2, \ldots, a_n\} \subseteq A$ and $V_{ai} \subseteq D_{ai}$, $i = 1, 2, \ldots, n$.

The interpretation of the above decision rule is intuitive. The conjunction of GO terms from the left-hand side of the rule describe a gene group that is represented by the value $v$. Induced decision rules may be used to describe biological function of genes composing a group of interest. Given object recognizes the rule if its attribute values satisfy the premise of the rule. The object supports the rule if it recognizes the rule and the decision given to the object is the same as the decision from the right side of the rule.

Due to the specific structure of the rule descriptors (GO terms) we could not use any of existing tools based on standard methods of induction of decision rules. Thus, considering the specific type of a data, we developed a rule induction algorithm that is suitable for our requirements [6].

## 3 System Architecture

RuleGO is an Internet application that makes the method of induction of decision rules for description purposes publicly available. The main core of the application – rule induction algorithm – is implemented in MATLAB® programming language due to the fact that we use some functions from MATLAB bioinformatics toolbox [8]. The rule induction algorithm is complex and depending on the algorithm input parameters, time of a single execution may vary from several minutes to several hours.

The main idea of the tool presented in this paper was to create a system for execution of existing rule induction application simultaneously in a distributed environment. As we did not want to interfere with the MATLAB application, we decided that the system should consist of two components: a MATLAB core and a suite of independent tools that are responsible for preparing input data, executing the rule induction application and postprocessing the results. Another important part of the system is a graphical user interface that allows the user to communicate with the system – it allows for submitting data and parameters of the rule induction algorithm and presents final results of analysis.

The architecture of the system is presented in the Fig. 1. We created the system on the based on the LGS4ns2 – distributed simulations environment that was previously designed and developed for the $ns-2$ simulator [4]. However, we adapted this solution to our requirements.

In the proposed approach, there is no centralized software that is responsible for managing, scheduling or executing tasks related to the submitted data. However there is possibility to configure a metascheduler that can supervise tasks workflow, execute tasks on selected clients, monitor the current state of the analysis and retrieve results of the computations. System clients can be located either on a main server or on remote machines and connect to the main server through network infrastructure. Such approach allowed us to create flexible, easily scalable, heterogeneous distributed environment. Client machines can be added to the running system or removed from it depending on the system load and available computational resources without any need to change system configuration scripts or restart the main server application.

After the list of genes is submitted to the system it is stored in the database with all other task-specific information such as the rule induction algorithm parameters and user or session identification needed for results presentation. Identifier of a specific task consists of an optional job name which can be provided by the user and a random number generated by the system. This identifier can be further used to retrieve information about current state of analysis.

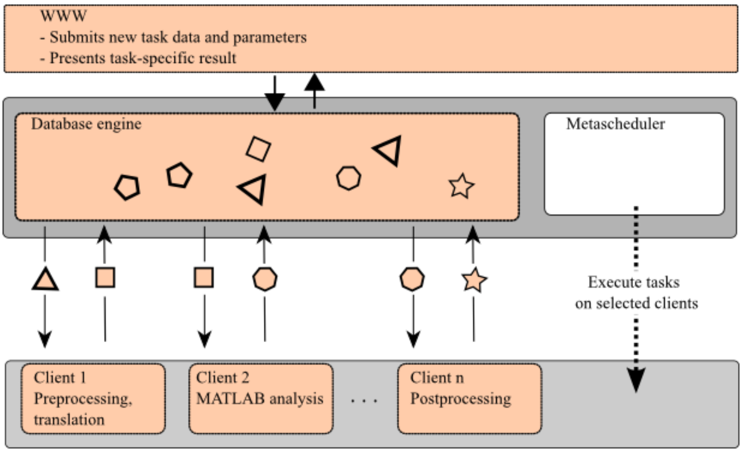



**Fig. 1.** The RuleGO system architecture. Data and algorithm parameters sent by the user are submitted into the database. Each client connects to the database in order to retrieve the client-specific data and submits the results of that data processing. Different geometric shapes symbolize the tasks with different types of parameters. Induced rules are presented to the user on RuleGO web page.

Clients work independently from each other. If the computational resources related to any client are currently available it connects periodically to the database in order to check if there is any task of the properties specific to that client type. Each type of task is identified by a set of unique attributes. The client retrieves from the database only the tasks that are assigned to that client type. If there is available data (specific to the client type and not processed by any other client) it is downloaded from the database, processed by the client and results are submitted back to the database.

In the system we can distinguish three types of clients that are responsible for the following tasks:

– **Data preprocessing.** This step involves translation of submitted gene list to a format accepted by rule induction application, its filtration and creation of reference gene, if needed.
– **Executing of MATLAB application.** This is the main part of the analysis. In this step, the list of translated gene identifiers is retrieved from the database together with all parameters of the rule induction algorithm and the MATLAB application is executed. If the analysis is finished, the resulting set of rules is stored in the database.
– **Results postprocessing.** This part includes additional analysis of the induced rules. The client retrieves a set of rules from the database in order to obtain names of genes supporting the rules. Then, PubMed journal database is searched for all papers that include symbols of these genes. The list of obtained publications is attached to each of the rules.

If needed, another type of a client can be included in the process of analysis. This can be simply done by defining new, client-specific set of parameters and introducing client application to the system environment.

### 3.1 Software Requirements

As it was mentioned, the main step of the rule induction analysis is performed by MATLAB application. Other parts of the system are implemented using standard open source tools and services. The part of the system responsible for preprocessing data, executing rule induction algorithm and postprocessing obtained rules is a Python script. For storing input data, algorithm parameters and results, we use PostgreSQL database management system. The GUI is based on PHP scripts executed as web pages by Apache 2.2 HTTP server. The operating system used for the server is Linux kernel 2.6.24-23.

We are not limited to any particular operating system that should be installed on a server or client machines. The only specific software requirement is related to the machines on which a rule induction application is executed – on such computers MATLAB software with the bioinformatics toolbox must be installed. In case we need to add any new client application to the process of analysis, we can use any programming language or technology.

## 4 User Interface

The system is freely available, there is no need to create a personal account in RuleGO system. However, after data submission the user is asked to provide an e-mail address. This e-mail address is stored for information purposes only.

Each user of RuleGO system has access to data submission form which is divided into two main sections: gene submission and rule induction algorithm parameters. In the gene submission part, the user can select an organism from the list of available species, submit a list of proper gene identifiers and a reference set of genes. In the rule induction section the user can provide the parameters for statistical test, choose the type of the gene ontology that will be used for creation of the decision table (biological process, molecular function or cellular component), select the minimal and maximal levels of the GO terms used, define minimal number of genes described by GO term, maximal number of rule descriptors and minimal number of genes supporting the decision rules. The user can also suggest a name of a job that will be further used to identify the results of the submitted task. The screenshot of submission form is presented in Fig. 2.

After submission, data preprocessing starts. Gene identifiers provided by the user are translated into the UniGene symbols [10] that are accepted by MATLAB application. The list of gene symbols acceptable to the service is available on RuleGO website. If some of the symbols could not be translated by a preprocessor, the list of invalid identifiers is presented to the user. After preprocessing is finished, execution of rule induction algorithm starts. A web page is generated which presents current state of computation and provides the results of computations after the analysis is finished. Such web page is unique and related to the


**Fig. 2.** Graphical User Interface – new task submission

specific computational task submitted by the user. The user may access this web page directly or find it by a job name.

The result includes a list of generated decision rules with additional information such as number of rules, coverage of the submitted gene set and other details of the experiment. Each obtained rule is presented with parameters such as accuracy, coverage, p-value, FDR, and quality measure value. For each rule following information also provided: a number of genes supporting and recognizing the rule, symbols of supporting genes, and a list of publications from PubMed database related to these genes.

### 4.1 Example of Analysis

In this section we briefly present the results of the analysis performed with the use of RuleGO application. We analyzed a group of 139 genes from one of the clusters described in paper [3]. Expression profiles of *Saccharomyces cerevisiae* genes were measured during several DNA microarrays experiments and divided into 10 groups based on recorded expression values. For one of these groups we computed decision rules.

We assumed following parameters for rule induction algorithm: hypergeometric test for over-representation of GO terms, significance level: 0.01, ontologies used: biological process, molecular function, cellular component, minimal ontology level: 3, minimal number of genes described by GO term: 3, maximal number of rule descriptors: 5, minimal number of genes supporting the rule: 2.

After analysis we obtained 12 decision rules covering 100% of submitted genes. Below, we present one of the generated decision rules:

```
rule No: 1
biopolymer biosynthetic process (sup=137)(rec=189)(lev=5)
cellular macromolecule biosynthetic process (sup=137)(rec=187)(lev=5)
rRNA metabolic process (sup=22)(rec=28)(lev=8)
ncRNA processing (sup=22)(rec=24)(lev=8)
ribosomal small subunit biogenesis (sup=10)(rec=10)(lev=6)
=> class is 1

Number of objects supporting the rule: 8
Number of objects recognizing the rule: 8
Acc:1.0  Cov:0.05755  PVal:0.00396   FDR:0.10246  Quality: 0.38299
Supp. genes: RPS11B  RPS14B  RPS11A RPS19A RPS0A RPS0B RPS19B RPS14A
```

## 5 Conclusions and Future Work

In this paper we presented a technical details of the RuleGO grid-based system architecture. The RuleGO system allows inducing decision rules that include GO terms in its premise. GO terms composing conditional part of the rules may be used for functional description of gene groups obtained as a result of DNA microarray experiment. We did not provided a detailed description of the rule induction algorithm as the main purpose of this paper was to present the architecture of the developed grid system.

The proposed system allows us to make publicly available the rule induction method. Due to the fact that the rule induction algorithm is complex and has high computational requirements we designed a grid-based system to ensure that the RuleGO application can handle simultaneous requests from its users. The proposed approach supports executing computational tasks in a dynamic, wide-area, heterogeneous distributed environment. The system architecture allows extending the process of analysis simply by defining new set of parameters and introducing new client type to the system.

Presented application is an example of the popular tendency to share methods and algorithms among researches all over the world. Another example of such Internet service is NetTRS system [9] developed in our Institute of Informatics that makes algorithms of induction of decision rules based on rough sets tolerance model [12] publicly available. However, due to the specific structure of the rule attributes (GO terms), we cannot use standard rule induction algorithms for gene groups description purposes.

The RuleGO service is still extensively developed. Future work will focus on extending functionality of the system – to reduce computational cost of the rule generation process, another rule induction algorithm based on heuristic method will be available. Other works will include development of the module that will allow the user to obtain clusters of genes on based on their expression values. We are also planning to extend functionality of the grid system by adding modules for tasks monitoring and system load measuring. This can be further used for improvement of the system performance.

## References


1. Ashburner, M., Ball, C.A., Blake, J.A., et al.: Gene ontology: tool for the unification of biology. Nat. Genet. 25, 25–29 (2000)
2. Baldi, P., Hatfield, G.W.: DNA Microarrays and Gene Expression. Cambridge University Press, Cambridge (2002)
3. Eisen, M.B., Spellman, P.T., Brown, P.O., Botstein, D.: Cluster analysis and display of genome-wide expression patterns. Proc. Natl. Acad. Sci. USA 95, 14863–14868 (1998)
4. Chrost, L., Brachman, A., Chydzinski, A.: A program suite for distributed simulation of networks in multi-core environments. IEEE Computer Society Press, Los Alamitos (2009)
5. Gruca, A.: Analysis of GO composition of gene clusters by using multiattribute decision rules. Biocybernetics and Biomedical Engineering 28(4), 21–31 (2008)
6. Gruca, A.: Charakteryzacja grup genów z wykorzystaniem reguł decyzyjnych. PhD thesis, Silesian University of Technology (2009)
7. Khatri, P., Drăghici, S.: Ontological analysis of gene expression data: current tools, limitations, and open problems. Bioinformatics 21, 3587–3595 (2005)
8. MATLAB: Bioinformatics ToolboxTM 3 Users Guide. MathWorks, Inc. (2008)
9. Michalak, M., Sikora, M.: Decision rule besed data models using NeTRS – system overview. In: Peters, J.F., et al. (eds.) Transactions on Rough Sets IX. LNCS, vol. 5390, pp. 145–156. Springer, Heidelberg (2008)
10. Pontius, J.U., Wagner, L., Schuler, G.D.: UniGene: A Unified View of the Transcriptome. In: NCBI Handbook. National Center for Biotechnology Information, Bethesda (2003)
11. RuleGO, http://rulego.polsl.pl
12. Skowron, A., Stepaniuk, J.: Tolerance approximation spaces. Fundamenta Informaticae 27, 245–253 (1996)

# On the Performance of AQM Algorithms with Small Buffers*

Łukasz Chróst, Agnieszka Brachman, and Andrzej Chydziński

Silesian University of Technology, ul. Akademicka 16, 44-100 Gliwice, Poland
{lukasz.chrost,agnieszka.brachman,andrzej.chydzinski}@polsl.pl

**Abstract.** Packet buffers in routers play a major role in congestion control in existing Internet. They compensate for the incoming traffic bursts transmitted by aggressive TCP applications. The appropriate sizing of buffers is important for providing equilibrium between the high link utilization, the loss ratio and the queueing delay. There is a growing need for routers with small buffers in high speed networks. In this paper we investigate the performance of active queue management algorithms with very small buffers. In particular, we show, that the use of very small buffers does not influence throughput on bottlenecked links if mixed (TCP / UDP) traffic is involved. On the other hand a very small buffer may lead to degradation of the inter-flow fairness.

## 1 Introduction

Network congestion control is a serious issue in modern Internet. The disproportion in available and required bandwidth leads to emersion of bottleneck links. This happens especially when TCP flows are considered, as TCP aims at full bandwidth utilization. A typical congestion avoidance mechanism consists of two elements: the TCP congestion avoidance algorithm and a set of network routers at bottlenecked links. The TCP congestion avoidance mechanism reacts to packet losses by reducing the sending rate, when the packet drops are performed by routers preceding the constricted links.

Queue management mechanisms, namely Droptail and various AQM (Active Queue Management) schemes, are employed by the routers to avoid a serious throughput decrease in the case of a mild congestion. The AQM mechanisms allow adjusting dynamically target queue length and packet drop probability and thus outperform Droptail FIFO mechanism in case of light and moderate congestions. However, their performance may vary significantly in different network scenarios. The usage of both Drop Tail and AQM schemes rises many new issues one of which is sizing the buffers.

The buffer size and the target queue size are two most important parameters considering queue management in routers. While the use of very large buffers

* This work is supported by the Ministry of Science and Higher Education under grant KBN N N516 381134.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 168–173, 2009.

lowers the packet drop probability, it also significantly influences packet transmission delays. Truncation of the buffer stabilizes the packet delays but it is believed to lead to bandwidth underutilization.

The BDP rule (Bandwidth-Delay Product) is a common method for buffer size calculation. Twice the bandwidth-delay product of the network is widely accepted as the sufficient queue length for both Droptail and AQM schemes. Raina and Wischik [1] showed, that the usage of small buffers is more efficient for wide range of TCP windows sizes. Gorinsky at al. [2] suggested, that queue limit of 2L datagrams, where L is the number of input links is sufficient, while Appenzeller et al [3] claimed that drastic buffer size reduction is possible in backbone routers. Wischnik and McKeown [4] proposed that even very small buffers (20–30 packets) could be sufficient, causing very small throughput reduction. Enachescu et al [5] later proved this claim through numerous simulations. Dhamdhere at al. [6] pointed out other issues concerning the use of small buffers, like high packet drop ratio and insisted on BDP-based buffer sizing.

We believe, that in order to solve this contradiction, more complex scenarios should be taken into consideration. Most of the aforementioned articles focus directly on the TCP-AQM interaction. The effect of the buffer truncation on UDP flows was omitted in the previous studies, while it is known that UDP-based applications can be more susceptible to high packet drop rate. We also show in this paper, that the inter-flow fairness declines, when very small buffers are used. Through numerous simulations we investigate the behaviour of various AQM schemes (namely RED, REM, PI, AVQ) operating with very small buffers (the target average queue length is set to 10 packets).

## 2 AQM Overview

Queue management is a very important element of the network infrastructure. It directly influences packet transmission delays. Implementing queue management algorithm to obtain a very low level of the very low buffer occupancy allows achieving high quality of offered services. However, it can lead also to the link underutilization. In this paper four different Active Queue Management (AQM) algorithms are evaluated along with the Drop Tail strategy. A short description of each mechanism is presented below.

The idea of RED (Random Early Detection) is to notify the TCP sender about incoming congestion by dropping packets before the buffer overflow occurs. RED calculates the current network load by counting the exponential moving average of the queue size – $avg$. If $avg$ is smaller than $min_{\mathrm{th}}$ threshold all the packets are enqueued, if $avg > max_{\mathrm{th}}$ all the packets are dropped. When $avg$ is between two thresholds, packets are dropped with linearly increasing probability. More accurate description of RED can be found in [7].

AVQ (Adaptive Virtual Queue) uses the idea of the virtual queue. A router with AVQ algorithm maintains a virtual queue whose capacity is less than or equal to the capacity of the link, $c$. To configure AVQ, the two parameters are required, namely $\gamma$ and $\alpha$ which are the desired utilization and damping factor, respectively [8].

REM (Random Exponential Marking) tries to regulate the queue length to a desired value $q_{\mathrm{ref}}$. It updates periodically the probability of dropping packets with step $\gamma$. The probability is calculated using parameter $\Phi$ [9].

The PI (Proportional Integral) controller uses the knowledge of the queue size to clamp the steady value of queue length to the specified reference value [10]. It uses the well known idea from control theory.

Although active queue management allows achieving the desired average queue length, the simple FIFO queue of fixed buffer size remains the most popular strategy applied at routers. The incoming packets are buffered until the queue is full. When there is no space left, packets are dropped.

## 3 Performance Evaluation

### 3.1 Simulation Scenarios and Topology

The ns-2 simulator was used to study AQM performance [11]. The standard bottleneck network topology, used also in this paper is shown in Fig. 1. All nodes are connected to the routers with 10 Mbps links. The link propagation delay is set to 1 ms. The link from router $R_1$ to router $R_2$ is a bottleneck for each connection. The link has the propagation delay of 1 ms and the bitrate 10 Mbps. This scenario is a scaled-down version of the "Data Center" from [12].

We investigated four AQM algorithms namely: RED, PI, REM and AVQ. In addition we compared the results with a simple FIFO (DT) queue. The simulations were run twice for each algorithm, setting a large and a small buffer. PI, REM and AVQ proved to be unable to maintain a low target queue length if a large buffer was available, thus they were tuned to operate on very short queues. All parameters used in the simulations are summarized in Tables 1 and 2.

Simulation traffic conforms to [12] and is composed of CBR, FTP and WWW connections. Three CBR, sixteen FTP and eight WWW connections exist on each sender node, the connections are established to the recipients located on nodes at the receiver side. The reverse path has almost 10% of background traffic consisting of nine CBR. Constant bitrate of the UDP flows is set to 125 Kbps, they use 1000-byte packet sizes. Over 50% of the flows perform bulk FTP transfer to the receivers. The remaining flows imitate http responses. The transmitted file

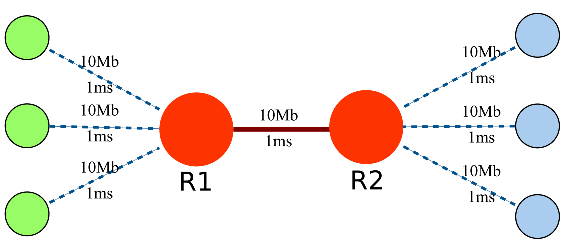



**Fig. 1.** Network topology

**Table 1.** Configuration parameters for AQM algorithms used in simulations with a small buffer

| Algorithm | Buffer size | Parameters changed |
|---|---|---|
| DT | 10 | — |
| RED | 20 | $min_{\mathrm{th}} = 5$ $max_{\mathrm{th}} = 15$ |
| PI | 10 | $q_{\mathrm{ref}} = 10$ |
| REM | 10 | $q_{\mathrm{ref}} = 10$, $\gamma = 0.001$, $\Phi = 1.001$ |
| AVQ | 10 | $\gamma = 0.98$ |

**Table 2.** Configuration parameters for AQM algorithms used in simulations with a large buffer

| Algorithm | Buffer size | Parameters changed |
|---|---|---|
| DT | 120 | — |
| RED | 120 | $min_{\mathrm{th}} = 30$ $max_{\mathrm{th}} = 60$ |
| PI | 120 | — |
| REM | 120 | $q_{\mathrm{ref}} = 50$, $\gamma = 0.001$, $\Phi = 1.001$ |
| AVQ | 120 | $\gamma = 0.98$ |

sizes and time intervals between consecutive transfers are set using generators described in [13]. FTP and CBR flows are active during entire simulation run. WWW traffic is generated between 40th and 45th second of simulation. Each simulation is 130 seconds long. Statistics are collected during the last 100 seconds.

### 3.2 Results Analysis

Tables 3 and 4 summarize results for FTP and CBR traffic when small and large buffers are used. The goodput achieved by FTP flows is comparable in both cases and for all algorithms. However, the difference in fairness among flows is significant. There is a huge disproportion in goodput experienced by TCP connections in the 1st case resulting in a very low Jain's index around 0.3. Only RED provides fairness for both FTP and CBR connections. It also provides the highest CBR goodput at the cost of higher drop rate for FTP connections. AQMs with a large buffer provide higher fairness for both FTP and CBR traffic. At the same time drop rate for CBR connections is twice as big as for FTP flows. The disparity is even bigger in a scenario with a small buffer.

The collected queue statistics are presented in Tables 5 and 6. The observed throughput is very close to the bottlenecked link capacity in both experiment groups. We believe that the introduction of 10% UDP (CBR) traffic is sufficient to diminish the underutilization of the bottleneck link. More, this suggests the need of compound scenarios to be used in place of TCP-only AQM experiments. The high packet drop rate, observable for all simulations is not distributed equally between the flows. The UDP drops are more frequent than TCP ones, especially if short buffers are used. The effect is evident in Tables 3 and 4.

**Table 3.** Results for AQMs with a small buffer

| | | DT | RED | AVQ | PI | REM |
|---|---|---|---|---|---|---|
| FTP | avg. thr [kbps] | 187.57 | 180.30 | 186.43 | 187.81 | 187.57 |
| | fairness | 0.29 | 0.92 | 0.32 | 0.35 | 0.29 |
| | avg. drop rate [%] | 8.65 | 19.86 | 8.02 | 8.46 | 8.65 |
| | fairness | 0.64 | 1.00 | 0.62 | 0.60 | 0.64 |
| CBR | avg. thr [kbps] | 78.45 | 111.04 | 84.29 | 76.98 | 78.45 |
| | fairness | 0.98 | 1.00 | 0.95 | 0.98 | 0.98 |
| | avg. drop rate [%] | 37.20 | 11.13 | 32.53 | 38.37 | 37.20 |
| | fairness | 0.93 | 1.00 | 0.80 | 0.95 | 0.93 |

**Table 4.** Results for AQMs with a small buffer

| | | DT | RED | AVQ | PI | REM |
|---|---|---|---|---|---|---|
| FTP | avg. thr [kbps] | 183.47 | 180.12 | 183.20 | 180.60 | 180.91 |
| | fairness | 0.96 | 0.97 | 0.95 | 0.97 | 0.96 |
| | avg. drop rate [%] | 11.39 | 17.64 | 11.41 | 11.97 | 16.80 |
| | fairness | 0.98 | 1.00 | 0.99 | 0.99 | 1.00 |
| CBR | avg. thr [kbps] | 88.82 | 113.01 | 90.89 | 103.24 | 102.02 |
| | fairness | 0.98 | 1.00 | 1.00 | 1.00 | 0.98 |
| | avg. drop rate [%] | 28.83 | 9.53 | 27.18 | 17.31 | 18.34 |
| | fairness | 0.88 | 1.00 | 0.89 | 0.94 | 1.00 |

**Table 5.** Results for queue management algorithm with small buffers

| | DT | RED | AVQ | PI | REM |
|---|---|---|---|---|---|
| Avg. queue size [pkts] | 8.43 | 8.97 | 9.05 | 9.46 | 8.43 |
| Throughput [Mbps] | 9.99 | 9.99 | 9.99 | 9.99 | 9.99 |
| Drop rate [%] | 12.54 | 18.75 | 10.93 | 11.74 | 12.54 |

**Table 6.** Results for queue management algorithm with large buffers

| | DT | RED | AVQ | PI | REM |
|---|---|---|---|---|---|
| Mean queue size [pkts] | 116.7 | 43.9 | 118.62 | 111.7 | 51.9 |
| Throughput [Mbps] | 9.99 | 9.99 | 9.99 | 9.99 | 9.98 |
| Drop rate [%] | 13.89 | 16.60 | 13.68 | 12.76 | 17.05 |

## 4 Conclusions

We have showed, that the use of very small buffers (10 packets) does not influence significantly the aggregated throughput on bottlenecked links if mixed (TCP / UDP) traffic is involved. Hence, **we believe that the throughput is not a sufficient criterion for the minimum buffer size evaluation**. The results suggest, that some other parameters, like the drop rate and inter-flow fairness are at least as important. The current AQM implementations, except for RED, are not suited for short queues, (in terms of maintaining a low target queue length) if operating with large buffers. Although the AQM schemes were capable of full bandwidth utilization in situation of low memory resources, they failed in terms of fairness and drop rate.

## References


1. Raina, G., Wischik, D.: Buffer sizes for large multiplexers: Tcp queueing theory and instability analysis. Next Generation Internet Networks, 173–180 (2005)
2. Gorinsky, S., Kantawala, A., Turner, J.: Link Buffer Sizing: A New Look at the Old Problem. In: Proceedings of 10th IEEE Symposium on Computers and Communications, 2005. ISCC 2005, pp. 507–514 (2005)
3. Appenzeller, G., Keslassy, I., McKeown, N.: Sizing router buffers. In: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications, pp. 281–292. ACM, New York (2004)
4. Wischik, D., McKeown, N.: Part I: buffer sizes for core routers. ACM SIGCOMM Computer Communication Review 35(3), 75–78 (2005)
5. Enachescu, M., Ganjali, Y., Goel, A., McKeown, N., Roughgarden, T.: Part III: routers with very small buffers. ACM SIGCOMM Computer Communication Review 35(3), 83–90 (2005)
6. Dhamdhere, A., Jiang, H., Dovrolis, C.: Buffer sizing for congested internet links. In: INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies, Proceedings IEEE, vol. 2, pp. 1072–1083 (2005)
7. Floyd, S., Jacobson, V.: Random early detection gateways for congestion avoidance. IEEE/ACM Transactions on Networking 1, 397–413 (1993)
8. Kunniyur, S., Srikant, R.: An adaptive virtual queue (AVQ) algorithm for active queue management. IEEE/ACM Transactions on Networking 12, 286–299 (2004)
9. Athuraliya, S., Li, V.H., Low, S.H., Yin, Q.: REM: active queue management. IEEE Network 15, 48–53 (2001)
10. Hollot, C.V., Misra, V., Towsley, D., bo Gong, W.: On designing improved controllers for aqm routers supporting TCP flows. In: Proceedings of IEEE Infocom, pp. 1726–1734 (2001)
11. The Network Simulator ns-2: Documentation, http://www.isi.edu/nsnam/ns/doc/index.html
12. Andrew, L.L.H., Marcondes, C., Floyd, S., Dunn, L., Guillier, R., Gang, W., Eggert, L., Ha, S., Rhee, I.: Towards a common TCP evaluation suite (2008)
13. Barford, P., Crovella, M.: Generating representative web workloads for network and server performance evaluation. SIGMETRICS Perform. Eval. Rev. 26(1), 151–160 (1998)

# Adaptive RED in AQM*

Joanna Domańska[1] and Adam Domański[2]

[1] Institute of Informatics Silesian Technical University,
Akademicka 16, 44-100 Gliwice, Poland
adam.domanski@polsl.pl
[2] Institute of Theoretical and Applied Informatics Polish Academy of Sciences,
Baltycka 5, 44-100 Gliwice, Poland
joanna@iitis.gliwice.pl

**Abstract.** The algorithms of queue management in IP routers determine which packet should be deleted when necessary. The active queue management, recommended by IETF, enhances the efficiency of transfers and cooperate with TCP congestion window mechanism in adapting the flows intensity to the congestion of a network. The article investigates the influence of the way of choosing packets to be dropped (end of the queue, head of the queue) on the performance, i.e. response time for in case of RED and ARED queues – two representative active queue management mechanisms used in Linux based IP routers.

## 1 Introduction

The algorithms of queue management in IP routers determine which packet should be deleted when necessary. The active queue management, recommended now by IETF, enhances the efficiency of transfers and cooperate with TCP congestion window mechanism in adapting the flows intensity to the congestion in the network. In this article we present analytical and simulation results and compare them with the behavior of this mechanisms in the real working routers. Our research was carried out in a laboratory environment. For the purposes of work the authors implemented RED and ARED mechanisms in the operating system Linux. Sections 2 and 3 gives basic notions on active queue management. Section 4 shortly presents analytical and simulation models of RED and ARED. Section 5 describes Linux IP router with AQM and discusses numerical results. Some conclusions are given in Sect. 6.

## 2 Active Queue Management

In *passive* queue management, packets coming to a buffer are rejected only if there is no space in the buffer to store them and the senders have no earlier

* This research was financed by Polish Ministry of Science and Higher Education project no. N517 025 31/2997 and supported by European Network of Excellence EuroFGI (Future Generation Internet).

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 174–183, 2009.

warning on the danger of growing congestion. In this case all packets coming during saturation of the buffer are lost. The existing schemes may differ on the choice of packet to be deleted (end of the tail, head of the tail, random). During a saturation period all connections are affected and all react in the same way, hence they become synchronised. To enhance the throughput and fairness of the link sharing, also to eliminate the synchronisation, the Internet Engineering Task Force (IETF) recommends *active* algorithms of buffer management. They incorporate mechanisms of preventive packet dropping when there is still place to store some packets, to advertise that the queue is growing and the danger of congestion is ahead. The probability of packet rejection is growing together with the level of congestion. The packets are dropped randomly, hence only chosen users are notified and the global synchronisation of connections is avoided. A detailed discussion of the active queue management goals may be found in [11].

The RED (Random Early Detection) algorithm was proposed by IETF to enhance the transmission via IP routers. It was primarily described by Sally Floyd and Van Jacobson in [9]. Its performance is based on a drop function giving probability that a packet rejected. The argument *avg* of this function is a weighted moving average queue length, acting as a low-pass filter and calculated at the arrival of each packet as

$$avg = (1 - w)avg + wq \tag{1}$$

where $q$ is the current queue length and $w$ is a weight determining the importance of the instantaneous queue length, typically $w \ll 1$. If $w$ is too small, the reaction on arising congestion is too slow, if $w$ is too large, the algorithm is too sensitive on ephemeral changes of the queue (noise). Articles [9,19] recommend $w = 0.001$ or $w = 0.002$, and [23] shows the efficiency of $w = 0.05$ and $w = 0.07$. Article [23] analyses the influence of $w$ on queueing time fluctuations, obviously the larger $w$, the higher fluctuations. In RED drop function there are two thresholds $min_{\mathrm{th}}$ and $max_{\mathrm{th}}$. If $avg < min_{\mathrm{th}}$ all packets are admitted, if $min_{\mathrm{th}} < avg < max_{\mathrm{th}}$ then dropping probability $p$ is growing linearly from 0 to $p_{\max}$ :

$$p = p_{\max} \frac{avg - min_{\mathrm{th}}}{max_{\mathrm{th}} - min_{\mathrm{th}}} \tag{2}$$

and if $avg > max_{\mathrm{th}}$ then all packets are dropped. The value of $p_{\max}$ has also strong influence on the RED performance: if it is too large, the the overall throughput is unnecessarily choked and if it's too small the synchronisation danger arises; [19] recommends $p_{\max} = 0.1$. The problem of the choice of parameters is still discussed, see e.g. [21,22]. The mean *avg* may be also determined in other way, see [20] for discussion. Despite of evident highlights, RED has also such drawbacks as low throughput, unfair bandwidth sharing, introduction of variable latency, deterioration of network stability. Therefore numerous propositions of basic algorithms improvements appear, their comparison may be found e.g. in [21]. In this article, we present analytical (based on Markov chain) and simulation models of RED and ARED. We assume either Poisson or self-similar traffic. Because of the difficulty in analyzing RED mathematically [24] RED and its modification ARED are studied in an open-loop scenario.

## 3 Adaptive RED

In this chapter we present different type of RED algorithms – ARED (Adaptive Random Elary Detection) – for this alghorithm the RED parameters are automatically adopted for network traffic. RED algorithm is very dependent on the selection of parameters. For the ARED algorithm parameters are subject to conditions on the network. The main purpose of the use of RED algorithm is to achieve a low delay for packets placed in the queue and obtain the maximum throughput of the link. In the traditional mechanism of RED the selection of appropriate parameter values in order to achieve this goal is extremely difficult. For the small network load or for the high-value parameter $p_{\max}$ the average value of the lenght of the queue oscillates around the minimum threshold $min_{\mathrm{th}}$. For the high network load or for the small-value parameter $p_{\max}$ the average value of the lenght of the queue oscillates around the maximum threshold $max_{\mathrm{th}}$ and often exceeds it. For the ARED alghorithm parameter $p_{\max}$ changes in the router operation, so that the average queue length is maintained between the $min_{\mathrm{th}}$ and $max_{\mathrm{th}}$. This approach reduces the problem of variability in the queue delays and minimizes the amount of rejected packets. The changes made to the RED algorithm [11]:

```
Every interval seconds:
   if(avg > target and pmax <= 0.5)
     increase pmax:
        pmax<-pmax + alfa
   else if(avg < target and pmax >= 0.01 )
            decrease pmax:
        pmax<-pmax *  beta

avg : the average queue length,
interval : the time at which calculations are carried out,
target : the factor, avg seeks to it,
alfa: increment factor,
beta  : decrease factor.
```

After the expiry of certain period of time (creator of the algorithm proposed 0.5 s.), algorithm checks the values of parameter target and $p_{\max}$. Parameter target is defined as:

$$[min_{\mathrm{th}} + 0.4 * (max_{\mathrm{th}} - min_{\mathrm{th}}), min_{\mathrm{th}} + 0.6 * (max_{\mathrm{th}} - min_{\mathrm{th}})] \tag{3}$$

If the average queue length exceeds the target and $p_{\max}$ is less or equal to 0.5, the parameter $p_{\max}$ is increased by a factor $\alpha$ defined as the lower value of 0.01 and $p_{\max}/4$, otherwise the $p_{\max}$ is reduced by a factor $\beta$ (authors of the algorithm proposed 0.9). The function of RED algorithm is to maintain the average length of the queue between min and max stick. The function of ARED algorithm is to maintain the average length of the queue between minimum and half of maximum. The $p_{\max}$ parameter changes between 0.01 and 0.5. Hence, the probability of rejection of the package for the $max_{\mathrm{th}}$ is between 0.01% and 50%. Correction of $p_{\max}$ is slowly. However, the algorithm needs 10 to 20 seconds to achieve the performance (Fig. 5).

## 4 Analytical and Simulation Models of RED and ARED

The RED or ARED queue mechanisms are represented by a single-server model based either on discrete-time Markov chain or simulation. The service time represents the time of a packet treatment and dispatching. Its distribution is geo metric. The model of incoming traffic was presented above. For both considered in comparisons cases, i.e. for geometric interarrival time distribution (which cor responds to Poisson traffic in case of continuous time models) and self-similar traffic, the considered traffic intensities are the same. A detailed discussion of the choice of model parameters is also presented in [22]. In Markov model, the Markov chain state is defined by the number of packets in the queue, the integer part of the avg value and by four flags u1, u2, u3, u4 approximating the rest of this value (as avg is a real number, it is impossible to attribute a state to each of the infinite number of its possible values) in the following way:

$$\frac{(i-1)*0.25+i*0.25}{2} \tag{4}$$

where $i$ is the number of non-zero flag. If all flags are null, we assume the integer value of avg. In case of self-similar traffic this state definition is supplemented by a variable denoting the state of the modulator. The vector $p$ of state probabilities is given by a system of linear equations

$$p = p * P$$

where $P$ is the transition probability matrix which is generally large (the number of states, hence the order of the matrix P may be hundreds of thousands or millions), sparse and ill conditioned, and the use of well known and broadly used numerical algorithms for algebraic and differential equation systems gives

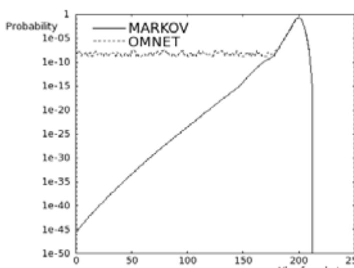



**Fig. 1.** Queue distribution for RED queue: geometric source, $\alpha = 0.5$, $\mu = 0.25$, $w = 0.07$, analytic and simulation results

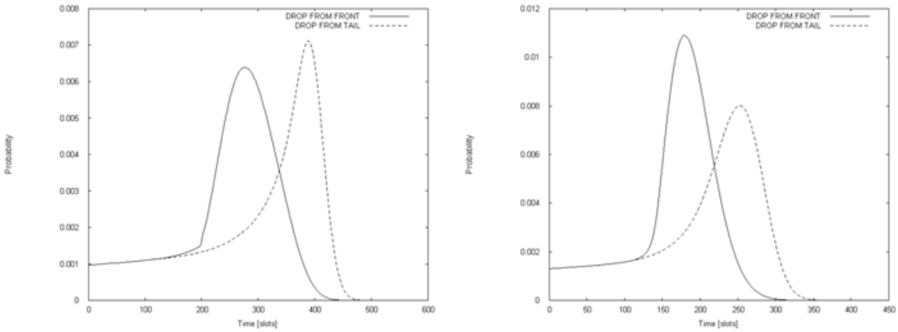



**Fig. 2.** Waiting times for RED (left) and ARED (right) queues: drop-from-front and drop-from-tail strategies, self-similar source, $\alpha = 0.5$, $\mu = 0.5$, $w = 0.07$, $\gamma = 0.07$

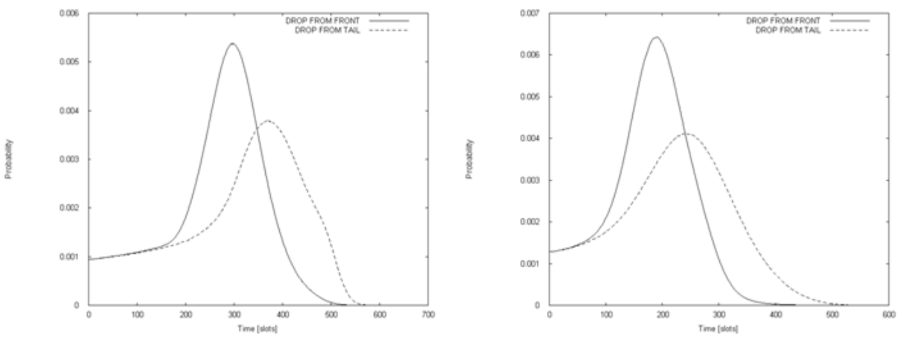



**Fig. 3.** Waiting times for RED (left) and ARED (right) queues: drop-from-front and drop-from-tail strategies, self-similar source, $\alpha = 0.5$, $\mu = 0.5$, $w = 0.07$, $\gamma = 0.5$

poor results. That is why a projection method using Krylov subspaces, as recommended in [23] was chosen. Our goal is to capture the influence of the way a packet is chosen to be deleted (end of the tail, head of the tail) on the RED and ARED queueing times. Input traffic intensity (for geometric and self-similar traffic) was chosen as $\alpha = 0.5$, and due to the modulator characteristics, the Hurst parameter of self-similar traffic was fixed to H $= 0.78$. The RED parameters had the following values: buffer size 250 packets, threshold values $min_{\mathrm{th}} = 100$ and $max_{\mathrm{th}} = 200$, $p_{\max} = 0.1$, $w = 0.002$ or $w = 0.07$. Parameter $\mu$ of geometric distribution of service times (probability of the end of service within a current time-slot) was $\mu = 0.25$ or $\mu = 0.5$. Due to the changes of $\mu$, two different traffic loads (low and high) were considered. In case of ARED policy, the traffic pattern and the buffer size are the same, parameters $Kl = min_{\mathrm{th}} = 100$ and $Kh = max_{\mathrm{th}} = 200$. The shaping parameter had three values $= 0.15, 0.5, 0.85$. Figure 1 displays a comparison of analytical and simulation results. They are almost identical if probabilities are greater then $10^{-10}$, for smaller values the simulation results are not significant (the simulation run involved 250 millions of packets) while Markov model is able to give probabilities of very rare events. If the mean queue length is relatively low, the influence of dropping scheme on queueing time is negligible: the introduction of drop-from-front strategy gives 0.7% shorter mean queueing time in case of RED and 0.6% shorter mean queueing time in case of ARED, see Fig. 1.

Naturally, the introduction of ARED gives shorter mean queue length and shorter mean queueing time compared to RED. However, when the Poisson traffic is replaced by self-similar one with the same intensity and preserving the same parameters of RED, the length of the queue grows and the influence of the dropping scheme is more visible: drop-from-front strategy reduces mean queueing time by 16.4%. A comparison of response time distributions for RED queue, for both strategies is presented in Fig. 2 (left). The same comparison in case of ARED queue is presented in Fig. 2 (right). In this case the response time with drop-from-front strategy is 18.1% shorter then for tail-drop mechanism. The change of wq value (from 0.07 to 0.002) in computation of moving av erage results in longer response time and mean queue, but the introduction of drop-from-front in place of tail-drop gives about 1% of changes. A comparison of queueing time distributions in these cases is given in Fig. 3. In case of heavy traffic, for both mechanisms RED/ARED, irrespective of the wq value and of the traffic self-similarity, drop-from-front strategy gives two times shorter mean queueing times.

## 5 Description of the Research

Figure 4 displays the experiments topology of the network. The Server is the most important part of the network. It works as a router with AQM algorithms (RED, ARED) implemented.

Correctness of operation of random early detection of congestion algorithms depends on correctly set up parameters. These parameters depends on conditions and kinds of flows in computer network. In our work we investigate different modification of RED algorithms. We also investigate the influence of the way packets are chosen to be dropped (end of the queue, head of the queue).

The RED parameters had the following values (presented below):

- size of queue $buf\_size = 25$ packets
- threshold $min_{\mathrm{th}} = 5$ packets
- threshold $max_{\mathrm{th}} = 15$ packets

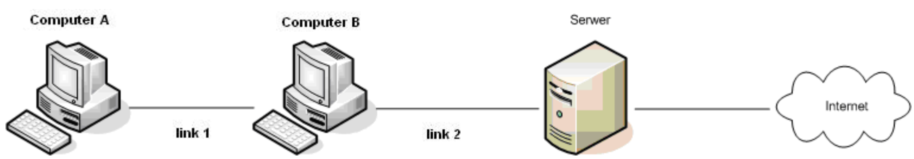



**Fig. 4.** Network used during the research

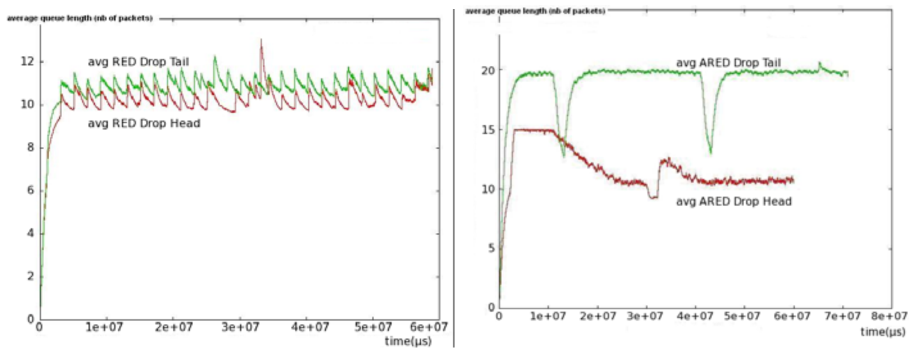



**Fig. 5.** Average queue length for RED (left), ARED (right) algorithm and TCP traffic

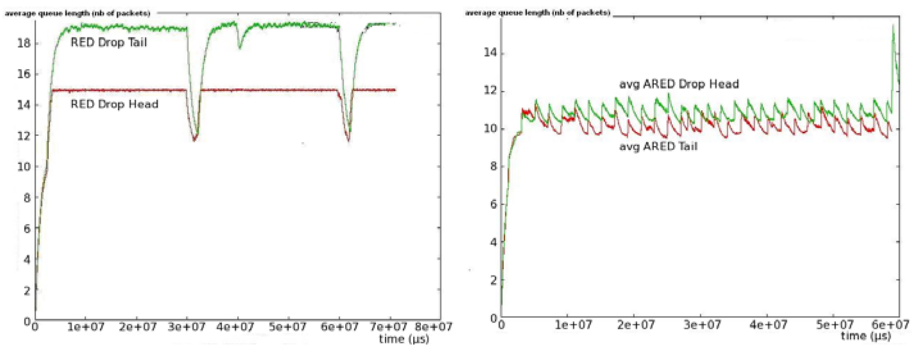



**Fig. 6.** Average queue length for RED (left), ARED (right) algorithm and UDP traffic

weight determining the importance of the instantaneous queue length $w = 0.002$ We set up parameters of RED based on values suggested in [9,7,17,18,19,20]. The ARED parameters had the following values:

- $wq = 0.0002$
- target $t = 0.5$

We observed: the mean length of the queue, waiting times and the number of dropped packets for three kinds of experiments:

- AQM behavior for TCP flows,
- AQM behavior for UDP flows,
- AQM behavior for mixed (TCP and UDP) flows.

Figure 5 shows the average waiting times in the queue for the TCP traffic. Figure 5 shows the same for UDP traffic. The main fuction of the ARED algorithm is to maintain the buffer fill between the upper and lower threshold. On Fig. 5 can be seen that ARED algorithm needs some time to start a valid

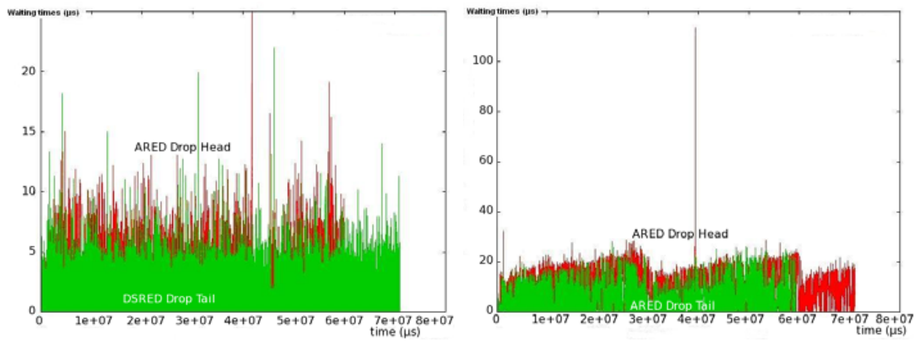



**Fig. 7.** Waiting times for ARED algorithm and TCP traffic (left), UDP traffic (right)

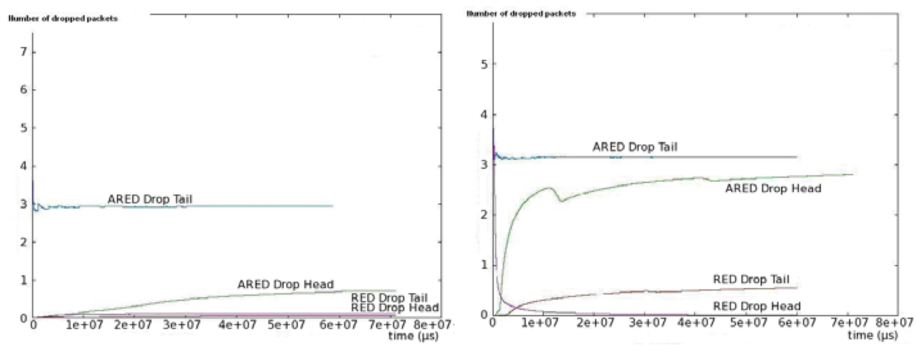



**Fig. 8.** Number of dropped packets for ARED algorithm and TCP traffic (left), UDP traffic (right)

work. At the beginning the average waiting time in the queue is larger, then fall and remain at a constant level. The creators of the algorithm indicate that the stabilization of the algorithm can take from 20 to 30 seconds [11]. Figure 5 shows good behaviour of ARED alghorithm for UDP traffic. Figure 8 shows that for the adaptive algorithm the number of lost packets is very small.

## 6 Conclusions

In this article we present the advantages of active queue management for Linux based routers. We have implemented two variants of RED algorithms. We also show the big influence of the way packets are chosen to be dropped (end of the queue, head of the queue) for the behavior of the router queue for UDP and mixed (TCP, UDP) traffic. Most of the articles evaluates AQM algorithms using analytical methods and simulation. In this article we additional present the behavior of this mechanisms in the real working routers. Our research was carried out in a laboratory environment. For the purposes of work the authors implemented, RED and ARED mechanisms in the operating system Linux. During the tests we analyzed the following parameters of the queue with AQM: the average waiting time package in the queue, the average length of the queue and number of rejected packages. Classical RED and ARED algorithms are suitable only for TCP protocols with congestion algorithm. For UDP traffic the examined parameters of the queue with AQM are significantly worse. In this article we also investigate the influence of the way packets are chosen to be dropped (end of the queue, head of the queue). When AQM algorithms drop packets from head of the queue it is possible to notice that the waiting times, the number of dropping packet and the average queue length are greater than for the case of drop packages from the end of the queue. Unfortunately for Drop From Front algorithms we have to make an additional operation: find and drop the first package in queue. In our router this operation is relatively slow due to executing in user space (not in kernel space) thus the results are worse than that for drop-from-tail strategy. Analytical and simulation studies ignore the impact of finding the first packet in queue operation on the effectiveness of RED algorithm. Therefore in these studies Drop From Front algorithm is better then Drop From End. We hope that better implementation of these algorithms in linux router will allow its work more effective.

## References

1. Domańska, J., Domański, A., Czachórski, T.: The Drop-From-Front Strategy in AQM. In: Koucheryavy, Y., Harju, J., Sayenko, A. (eds.) NEW2AN 2007. LNCS, vol. 4712, pp. 61–72. Springer, Heidelberg (2007)
2. Kapadia, A., Feng, W., Campbell, R.H.: GREEN: A TCP Equation Based Approach to Active Queue Management, http://www.cs.dartmouth.edu/~akapadia/papers/UIUCDCS-R-2004-2408.pdf
3. Athuraliya, S., Li, V.H., Low, S.H., Yin, Q.: REM: Active Queue Management, http://netlab.caltech.edu/FAST/papers/cbef.pdf
4. Kunniyur, S.S., Srikant, R.: An Adaptive Virtual Queue (AVQ) Algorithm for Active Queue Management, http://comm.csl.uiuc.edu/srikant/Papers/avq.pdf
5. Hashem, E.: Analysis of random drop for gateway congestion control, http://www.worldcatlibraries.org/oclc/61689324
6. Feng, W., Kandlur, D., Saha, D., Shin, K.: Blue: A New Class of Active Queue Management Algorithms, http://citeseer.ist.psu.edu/feng99blue.html
7. Feng, W.H., Kandlur, D.D., Saha, D., Shin, K.G.: A Self-Configuring RED Gateway, http://citeseer.ist.psu.edu/470052.html
8. http://en.wikipedia.org/
9. Floyd, S., Jacobson, V.: Random Early Detection gateways for Congestion Avoidance, http://www.cs.ucsd.edu/classes/wi01/cse222/papers/floyd-red-ton93.pdf
10. Random Early Detection (RED): Algorithm, Modeling and Parameters Configuration, http://photon.poly.edu/~jefftao/JTao_RED_report.pdf
11. Floyd, S., Gummadi, R., Shenker, S.: Adaptive RED: An Algorithm for Increasing the Robustness of REDs Active Queue Management, http://citeseer.ist.psu.edu/448749.html

12. Lin, D., Morris, R.: Dynamics of Random Early Detection, https://pdos.csail.mit.edu/~rtm/papers/fred.pdf
13. RFC 793 – Transmission Control Protocol, http://www.faqs.org/rfcs/rfc793.html
14. http://en.wikipedia.org/wiki/TCP_congestion_avoidance_algorithm
15. Alemu, T., Jean-Marie, A.: Dynamic Configuration of RED Parameters, http://citeseer.ist.psu.edu/728472.html
16. Verma, R., Iyer, A., Karandikar, A.: Towards an adaptive RED algorithm for archiving dale-loss performance, http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1214606
17. Yang, X., Chen, H., Lang, S.: Estimation Method of Maximum Discard Probability in RED Parameters, http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1712588
18. Hong, J., Joo, C., Bahk, S.: Active queue management algorithm considering queue and load states, http://ieeexplore.ieee.org/Xplore/login.jsp?url=/iel5/9617/30391/01401608.pdf
19. Floyd, S.: Discussions of setting parameters (1997), http://www.icir.org/floyd/REDparameters.txt
20. Zheng, B., Atiquzzaman, M.: A framework to determine the optimal weight parameter of red in next generation internet routers, The University of Dayton, Department of Electrical and Computer Engineering, Tech. Rep. (2000)
21. May, M., Bonald, T., Bolot, J.: Analytic evaluation of red performance. In: IEEE Infocom 2000, Tel-Aviv, Izrael (2000)
22. Feng, W.C., Kandlur, D., Saha, D.: Adaptive packet marking for maintaining end to end throughput in a differentiated service internet. IEEE/ACM Transactions on Networking 7(5), 685–697 (1999)
23. May, M., Diot, C., Lyles, B., Bolot, J.: Influence of active queue management parameters on aggregate traffic performance, Research Report, Institut de Recherche en Informatique et en Automatique, Tech. Rep. (2000)
24. Hassan, M., Jain, R.: High Performance TCP/IP Networking. Pearson Education Inc., London (2004)

# QoS Aware MPLS Multicast in the MAN DiffServ Domain

Slawomir Przylucki

University of Economics and Innovations,
7/9 Melgiewska Str, 20-209 Lublin Poland
spg@spg51.net

**Abstract.** The continuous development of new Internet technologies and multimedia services tend to an ever greater interest in practical solutions of the multicast transmission. Simultaneously, service providers more and more pay attention on appropriate selection of mechanisms to guarantee the QoS parameters needed for the proper delivery of services sensitive to transmission parameters. From a practical point of view, the final solution for the QoS aware multicast transmission should be on the one hand, closely associated with existing standard networks mechanisms on the other hand, be sufficiently flexible to offer proper level of independence from specific technologies. These, for the first look, conflicting requirements can be solved by the packet labeling technique. This raises the question whether the current operator of the wide or metropolitan area network may implement multicast transmissions with a guarantee of the Quality of Service (QoS) parameters based on Multiprotocol Label Switching (MPLS) technology. The answer to this question is the content of this article.

## 1 Introduction

To transfer data in multicast technology it is necessary to apply specific algorithms and protocols. These algorithms are responsible for determining tree distribution while the protocols for the delivery of packages to the group recipient. On the other hand, to improve the quality of transmission in today's Internet, a method to guarantee QoS parameters has been developed. One of the most popular mechanisms to ensure QoS is the service differentiation and based on this principle DiffServ network architecture.

The opportunity for relatively simple and scalable combination of these two networks solutions is Multiprotocol Label Switching (MPLS). Label switching significantly improves performance and scalability of the network backbone. From the point of view of the topic of presented article, the main attention is paid on the traffic engineering extension of MPLS technology referred to as MPLS Traffic Engineering (MPLS-TE). That extension seems to be promise solution for straightforward realization of multicast transmission in the MPLS-DiffServ domain.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 184–191, 2009.

### 1.1 Traffic Engineering

A traffic engineering in packet networks includes the measurement, modeling, characterization and control of traffic and use of packages to achieve specified performance objectives, including: a fast and reliable packets transfer, efficient use of available network resources and network capacity planning. The main purpose of the use of the Traffic Engineering (TE) is to minimize or even avoid the occurrence of bottlenecks in the network during transmission of data. Other interesting feature of the TE extension is the possibility to combine packet marking with QoS mechanism. The obvious example of that could be a MPLS-TE mechanism [2].

### 1.2 The Use of MPLS-TE in MPLS Domain

To form a Label Switched Path (LSP) and in the next step the MPLS-TE tunnel, the basic MPLS configuration must be present on the routers. Moreover, the link-state protocol must be configured in order to each router can have information about the status of all links. They are later distributed to all routers in the area, creating a form of TE database. Based on information in the TE database, Path Calculation Algorithm (PCALC) or Constrained Short Path First (CSPF) calculates the shortest and best route with the required parameters from the router "head" to "tail" [3], [9]. Based on the TE database, the shortest LSP path is calculated and reserved. In case of TE extension, that path is the shortest LSP path with specific resources required for the tunnel (e.g. required bandwidth). Usually, the reservation is done with PATH and RESV messages from the Resource ReSerVation Protocol (RSVP) protocol. From head to tail of the tunnel, the RSVP PATH message is to be sent. It transfers the request for an MPLS label and temporarily allocates resources along the path. As the response on the PATH message, the RSVP RESV message is sent back to the head. That message contains the label and says to all routers through which passes, to set the resources required for the links, which will use the tunnel [9], [12], [7]. A TE tunnel is configured on the head Label Switch Router (LSR) in the two possible ways [7]:

- Explicitly – all routers through which the tunnel will be passed, has to be pointed out, together with LSR "head" and LSR "tail";
- Dynamic – the LSR "tail" router of the tunnel is to be only indicate, the head LSR router selects the best LSP path to the destination.

When more than one TE tunnel is defined, and we want to chose one of them, then the tunnel priority must be introduced. The lower assigned priority value, range from 0 to 7, gives the tunnel a higher priority and thus it is chosen as the prefered route for the data transmission. Inside the MPLS-TE domain, the TE tunnel can be attributed to two different types of the priorities [7]:

- Setup priority – defines the importance of the tunnel compared with the other;
- Holding priority – indicates the importance of the tunnel from the reservation maintaince point of view.

## 2 Multicast Transmissions in the MPLS Domain

MPLS technology offers great flexibility in the packages transmission by splitting the mechanisms responsible for the data transfer path from the control components. However, this advantage does not improve the functionality of IP multicast routing. The primary problem arise when multicast trees at third ISO OSI Reference Model (OSI) layer is to be mapped on multiple MPLS LSP paths.

### 2.1 Using RSVP-TE in MPLS Technology

MPLS multicast transmission is based on the hybrid model. In this model, RSVP tunnels lead distributions of the labeled packages inside the MPLS domain, and the multicast tree, formed by the PIM protocol, are dynamically created outside the MPLS domain.The TE tunnels are creating using the previously mentioned RSVP protocol [5].

One-way TE tunnel begins in the “ingress” LSR, and ends on the “egress” LSR. In case of multicast, may be several or more Point-to-Multipoint (P2MP) tunnels extending from the “ingress” LSR to multiple “egress” routers. P2MP LSP is made up of so-called sub-LSP and each of them is indicated by self PATH and RESV RSVP messages [8], [14]. In addition, multicast require that PATH and RESV messages contain the new object, P2MP session object. Thanks this object routers know which sub-LSP is part of what LSP. The tunnel is a static, pre-determined part of the multicast tree inside the Protocol-Independent Multicast (PIM) enabled MPLS domain. Outside the domain, multicast trees are created dynamically in response to Internet Group Management Protocol (IGMP) reports generated by the recipients wishing to join a specific multicast group. Recipients, which finally receive stream of packets, create the multicast

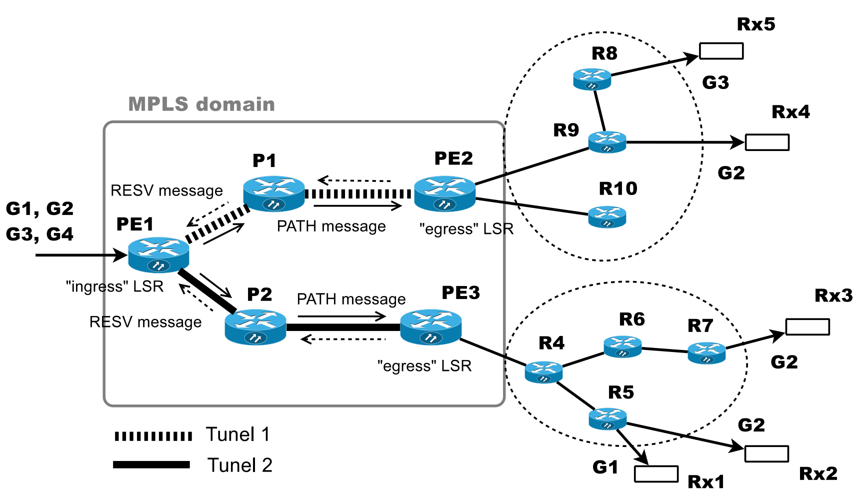



**Fig. 1.** The hybrid model of a multicast transmission

islands attached to the "egress" LSR [13], [11]. Figure 1 presents a process of the creation of P2MP trees using RSVP-TE.

The tunnel extends from "ingress" LSR router PE1 to "egress" LSRs routers PE2 and PE3, which are attached to the multicast recipients island. To set the tunnel, router PE1 sends a PATH message to the "egress" routers and as the response, the RESV messages with a label are sent. Router PE1 is attached to the multicast groups G1, G2, G3. Tunnels inside the MPLS domain distribute packets belonging to these groups among to "egress" LSR routers. When the receiver RX2 and RX4 wish to receive a packages from the G2 group, they send IGMP messages to the routers to which they are linked (R5, R9) and by this mean signalling the willing to attach to the group. In next step, these routers generate a PIM Join message. Thus the the packets start to flow from the source router G2 to PE1. Here comes the process of packets replication and thanks to that they can be send further along tunnels 1 and 2 to routers PE2 and PE3. From that point, the final recipients RX4 and RX2 are easyly reached [5].

## 3 MPLS Labeling in the DiffServ Domain

Network traffic inside IP DiffServ domain is divided into classes based on the assignment of different values in the modified ToS field, called Differentiated Services Code Point (DSCP) field, of an IP packet header. In this way the package is treated differently by each node, depending on the defined Per Hop Behaviour (PHP). To be able to apply similar rules in the MPLS domain, packets belonging to the class must be determined by means of a label or other value inside MPLS header.Other words, some way of maping between DSCP and label has to be defined. Thus, two models were developed to allow the use of quality of service classes in DiffServ MPLS domain [2], [10]:

- EXP-inferred-class LSP – E-LSP,
- Label-inferred-class LSP – L-LSP.

E-LSP model uses only three-bit EXP field inside MPLS labels to map the rules of PHP. Consequently, it can simultaneously transmit up to eight classes of service. Stream of packets entering the MPLS-DiffServ domain is switched based on the value of the label, which specifies the router of the next hop, and the type of service is defined as the EXP field value [2]. The E-LSP solution can not be used in an ATM technology, since ATM does not apply MPLS labels. (labels a coded by VCI and VPI values).

According to the L-LSP solution, the service type is defined by the value of labels in conjunction with corresponding EXP field. This allows transmission of only one DiffServ class inside one LSP path (one class corresponds to a class of Forwarding Equivalence Class). This approach also requires signalling protocol together with the specific extensions that identify the LSP as an L-LSP and identify the class, which L-LSP corresponds to. Thus, PHB-EXP mapping can not be defined statically as it is a case in the E-LSP approach [10]. Moreover, in this model, all packets belonging to the AF class must be sent along the same LSP path. The Assured Forwarding (AF) class packets are placed in a common queue, and differ only in the probability of rejection and the order can not be changed. Thus, packages can not be assigned to different labels, which forces a same transmission route. In that case the likelihood of rejection of the package determines the EXP (or Cell Loss Priority CLP field in the ATM) [10]. Although the value of EXP field can be changed many times, the DSCP value on the "egress" LSR router is the same as at the entrance to the MPLS domain. That feature is kind of tunnelling. Three models have been defined to allow such tunneling [2]:

- pipe model,
- short pipe model,
- uniform model.

In the pipe model, the "ingress" LSR router copies bits from DSCP field to EXP or EXP fields id defined statically. The "egress" LSR router treats packages based on the MPLS principles and the MPLS EXP bits are not maped on the DSCP field.

The short pipe model is similar to the pipe one with one difference: "egress" LSR router treats packets according to PHB rules defined in the DSCP and EXP bits are not transferred to the DSCP field.

In contrast, in the uniform model, "ingress" LSR router needs to copy bits of the DSCP field for the EXP and the "egress" LSR router copies a value of EXP field for DSCP field. This approach makes the package all the time to be one and the same class [3], [6].

MPLS TE and DiffServ can operate independently on the same network, providing the background for unicast and multicast transmission. The models of the MPLS tunneling inside DiffServ are presented in Fig. 2 [6]. In the case of the DiffServ-Aware traffic engineering, each tunnel is set up with the restriction for the class of service and defined pool of bandwidth. If there are no statically defined bandwidth pools, the global maximum throughput of the link is used. One can remember that pools also can be defined independently from each other or arranged in a stack, one on the other. Model, in which pools are arranged in a stack is called Russian doll model (RDM). Also in literature is described the other model, called maximum allocation model (MAM). In the MAM model pools are independent of each other and are statically and have clearly defined boundaries of available bandwidth [10].

It is also possible to transmit packages divided into different classes of services in different TE tunnels. In principle, the choice of the tunnel is based on the checking the value of EXP or DSCP field of incoming packet. Unfortunately, there is still a limit of the three-bit EXP field which allows only eight classes of QoS, so you can use a maximum of eight tunnels in order to map each class of traffic to specific tunnel [14].

At this point,one has to remember that so far there are no standards for describing the solutions that combine DiffServ and multicast technologies. So far, temporary solutions are used, unfortunately, given a number of restrictions. The most promising among the proposed solutions are: DSMCast and EBM. Due

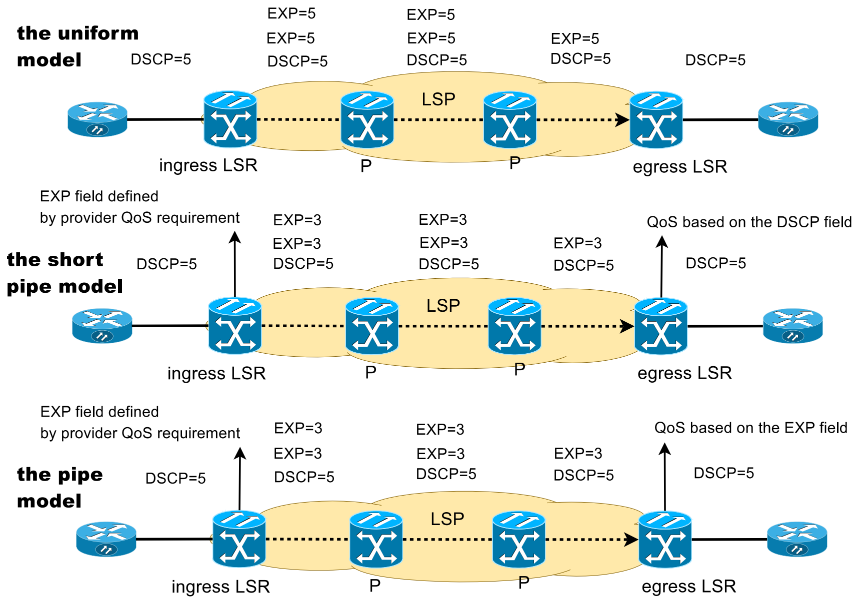



**Fig. 2.** MPLS-DiffServ Tunneling Models: the uniform model at the top, the short tube tube model inside and the pipe model at the bottom

to these facts, the use of MPLS-TE in combination with Diffserv architectures seems to be so interesting. The proposed solution for a typical MAN network is presented in the next part of article.

## 4 Straightforward Realization of QoS Guarantees in DiffServ-MPLS Domain

In order to verify the operation of the techniques described in earlier parts of the article, the prototype DiffServ domain together with cooperating MPLS-DiffServ domain was developed and tested. The very typical set of equipment were used for this purpose, a set of classic Cisco routers (1700 series routers and 2800) and the Linux system (CentOS version 5.2). That selection of hardware components allows an assessment of whether the guarantee of the parameters with the use of combined techniques MPLS and DiffServ is a stable solution, available to the wide range of the Internet service providers. Test included the E-LSP technique with different pipes‘ models. The MAM model was utilized for bandwidth assignation. Reservation protocol during multicast tests was RSVP. While the unicast transmission fulfilled all the expectations resulting from the utilized network technologies. serious difficulties arise in the case of multicast transmission.

While no problems were observed in the backbone of a network using techniques MPLS-TE to control QoS parameters, in the network edge using the

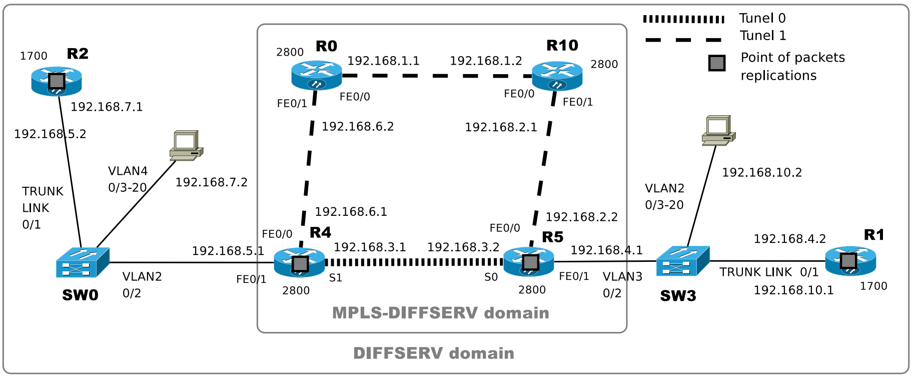



**Fig. 3.** Prototype MPLS-DiffServ network

differentiation of services obtained results were at first attempt questionable. One needs to keep in mind that in practical solutions of service providers networks, the labeling is a technique that is not supply the final customer. It is therefore necessary to take into account the situation that the access network is built on the basis of the DiffServ or even IntServ [4]. In this case, it was necessary to control the placements of the packets‘ replication point. These points can not be kept at any router within the DiffServ domain. Their location should allow the replication before the classification in DiffServ edge routers takes place. Than is illustrated in the Fig. 3 where the points of the packets replication are clearly point out. Such a solution, albeit with limited scalability, is the practical application of the concept of Edge-based Multicasting EBM [1]. The structure of prototype network is presented on the Fig. 3. During testing, network traffic in the DiffServ domain was differentiated on the basis of port of destination, and thus divided into 2 classes: “Platinum” and “Gold”. Then the packages were sent to the MPLS-DiffServ domain, where the input interface of the router R4 labeled them according to E-LSP method (ip2mplsin policy) while the output interface used ipmplsout policy. For the packets belonging to a class “Gold”, the EXP field was set to 5, and while packages classified as Class “Platinum” were assigned the value of EXP 3. Sample configurations of whole prototype network is available to all interested.

## 5 Conclusions

Constantly increasing demand for efficient multimedia services tends to need to introduce the concept of QoS guarantee for multicast transmissions. Designed and configured prototype network proved that in the case of typical local Internet service providers, MPLS and Diffserv techniques can be combined in a fairly easy manner. In this way, one can effectively ensure network parameters and their variability during the unicast as well multicast transmission. In a small network with few customers and not so complicated structure these techniques can work out of box. However, in the case of joining the multicast group at inner points of DiffServ domain, guarantees for traffic parameters can not be easyly fulfilled So far, there are some promising proposals for solving this common problem, among them an Edge technique [1]. There is also tested solution based on a hybrid MPLS-TE multicast. The developed prototype network combines both solution and proved that they are available on the existing network equipment. Scalability of the proposed solution pretend it to be used in big campus networks and typical metropolitan area network. Concluding, in the case of multicast transmission requiring certain guarantee of network resources, traffic engineering is becoming the most important advantage of MPLS technique.

## References

1. Striegel, A., Bouabdallah, A., Bettahar, H., Manimaran, G.: EBM: A New Approach for Scalable DiffServ Multicasting. In: Stiller, B., Carle, G., Karsten, M., Reichl, P. (eds.) NGC 2003 and ICQT 2003. LNCS, vol. 2816, pp. 131–142. Springer, Heidelberg (2003)
2. Alvarez, S.: QoS for IP/MPLS Networks. Cisco Press, Indianapolis (2006)
3. De Ghein, L.: MPLS Fundamentals. Cisco Press, Indianapolis (2006)
4. Ferguson, P., Huston, G.: Quality of Service: Delivering QoS on the Internet and in corporate networks. John Willey and Sons Ltd., England (2001)
5. Minei, I., Lucek, J.: MPLS-Enabled Applications Emerging Developments and New Technologies. John Wiley and Sons Ltd., London (2005)
6. Gallaher, R.: MPLS Training Guide – Building Multi Protocol Label Switching Networks. Syngress Publishing, Rockland (2003)
7. Osborne, E.: Traffic Engineering with MPLS. Cisco Press, Indianapolis (2002)
8. Aggarwal, R., Yasukawa, S.: Extensions to Resource Reservation Protocol – Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs). RFC4875 (2007)
9. Awduche, D., Berger, L.: RSVP-TE: Extensions to RSVP for LSP Tunnels. RFC 3209 (2001)
10. Davie, B., Wu, L., Davari, S.: Multi-protocol Label Switching (MPLS) Support of Differentiated Services, RFC 3270 (2002)
11. Estrin, D., Farinacci, D., Helmy, A., Thaler, D.: Protocol Independent Multicast – Sparse Mode(PIM-SM): Protocol Specification, RFC 2117 (1997)
12. Kompella, K., Rekhter, Y.: Signalling Unnumbered Links in Resource ReSerVation Protocol – Traffic Engineering (RSVP-TE), RFC 3477 (2003)
13. Ooms, D., Sales, B., Livens, W., Acharya, A., Griffoul, F., Ansari, F.: Framework for IP Multicast in MPLS, RFC 3353 (2002)
14. Yasukawa, S.: Signaling Requirements for Point to Multipoint Traffic Engineered MPLS Label Switched Paths (LSPs), RFC 4461 (2006)

# ServeR: .NET-Based Infrastructure for Remote Services of Statistical Computing with R-Project

Dariusz Rafał Augustyn and Łukasz Warchał

Silesian Technical University, ul. Akademicka 16, 44-100 Gliwice, Poland
draugustyn@polsl.pl

**Abstract.** The paper shows the efficient .NET-based system of the remote access for statistical computing methods provided by the R project functionality. The system can be integrated with thick or thin client applications using Windows Communication Foundation. In the batch mode a task (represented by a packet bundle of R scripts and input data files) is sent to the communication server (the web application hosted by the Internet Information Services) and then placed in a queue hosted by MSMQ. The computing server (hosted by Windows Service) executes the task using R.exe module and sends back the response (packet bundle of stdout content, output data files and graph ones). The middle layer MSMQ supports asynchronic handling of long running tasks. The system supports online mode – the remote interactive R session. The horizontal system scalability is enabled by multiplying computation servers spanned over network nodes.

## 1 Introduction

Modern information systems should be supported by business intelligence extensions and analysis ones for managers and scientists. Statistical data analysis can be enabled by using some module of statistics engine. Very popular, modern and functionality advanced one is the multiplatform R project [1,2].

Statistics analysis for healthcare information systems databases are especially required by medical scientists who are familiar with R functionality. This paper presents multilayer and multiuser system, called *SeverR*[1], designated for efficient and easy exposing R functionality. The functional scope of the system was agreed with an ordering software company developing medical systems. There are some customer key requirements which are fulfilled by *ServeR*, e.g.:

- system architecture based on .NET technology,
- connectivity technology based on Windows Communication Foundation (WCF) [8],
- both thick and thin client application enabled (WinForm and ASP.NET [9] enabled),

[1] This work has been supported by fund no. BNI/0004/08 from The Polish Agency for Enterprise Development (PARP Page: http://en.parp.gov.pl).

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 192–199, 2009.

– separate software layers: for client requests handling (a communication server) and for statistics computation processing (a calculation server); enable accepting requests by communication server, even when calculation server is down or busy,
– enable handling long running statistic R scripts (for disconnected clients too),
– enable remote R session mode (remote R console functionality),
– enable local and remote files and remote relational database as an input data for R scripts,
– logging user and system activities,
– sharing (committed) R scripts,
– scalability feature (e.g. by increasing number of computational servers).

## 2 Known Approaches

The problem of exposing R functionality can be solved using open standard of web services, where web methods map to a defined list of R functions. Such approach is commonly used (e.g. [3]). Some disadvantage of this approach for given requirements is limited hardcoded list of web methods which is inflexible for undefined user tasks (ad'hoc defined).

The *Rweb* approach [4] based on Perl/CGI technology, enables a user to upload R code and input data files and then obtain results. The web application can be used for remote executing simple R scripts.

The *Rwui* approach [5,6] lets create ready-to-use simple web applications based on R executable module in backend. The system enables interactive defining web application GUI, uploading R coded logic and finally downloading automatically generated java-based application code in a war file, ready for deploying on some JEE Tomcat Web Server.

Because some requirements (e.g. easy .NET application integration, .NET technology usage, ad'hoc R-script logic definition, ownerships of source codes) those approaches wasn't used, but some elements of *ServeR* concept are similar to above-mentioned ones.

## 3 ServeR Architecture

The system architecture will be explained by showing architecture of three main components:

– client application proxy object – (*CliApp*),
– web-based communication server (*CommSvr*) hosted by MS Internet Information Services,
– R-based calculation server (*CalcSvr*) hosted by MS Windows services.

Microsoft Message Queuing (MSMQ) system should be used for connectivity between the *CommSvr* and *CalcSvr*. Such layer separation lets these servers act independently (especially it accepts client requests by *CommSvr*, even when none *CalcSvr* is currently running).

Both connectivity channels (between *CliApp* and *CommSvr*, between *CommSvr* and *CalcSvr*) are defined and implemented using WCF technology. It enables to choose a proper transport mechanism like: named pipes, web services, msmq messages [8,9]. The component diagram for *ServeR* is shown on Fig. 1.

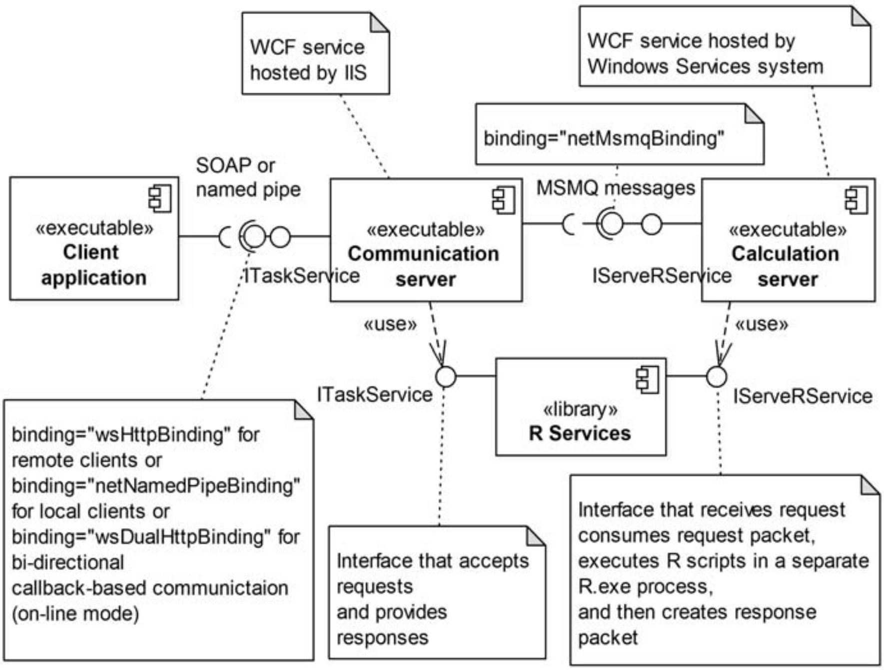



**Fig. 1.** *ServeR* component model

## 4 Client Application and Communication Server

The deployment diagram for *CliApp* and *CommSvr* is shown on Fig. 2. Two technological variants of client application are presented. Thick *CliApp* (e.g. WinForms based) communicates with *CommSvr* using web services (in deployment specification, in client element, binding attribute is set to "wsHttpBinding"). Thin *CliApp* (ASP.NET application) communicates using named pipes (binding attribute is set to "netNamedPipeBinding"). This fast method of interprocess communication is possible if *CliApp* and *CommSvr* are run on the same machine. *CommSrv* hosted by IIS accepts requests from both type of clients (two endpoint definitions in deployment specification).

A request from *CliApp* (a stream of zipped package containing R-scripts and input data file) is accepted by *CommSvr* and passed to MS queuing system. A response (a stream of zipped package containing result stdout, output data files,

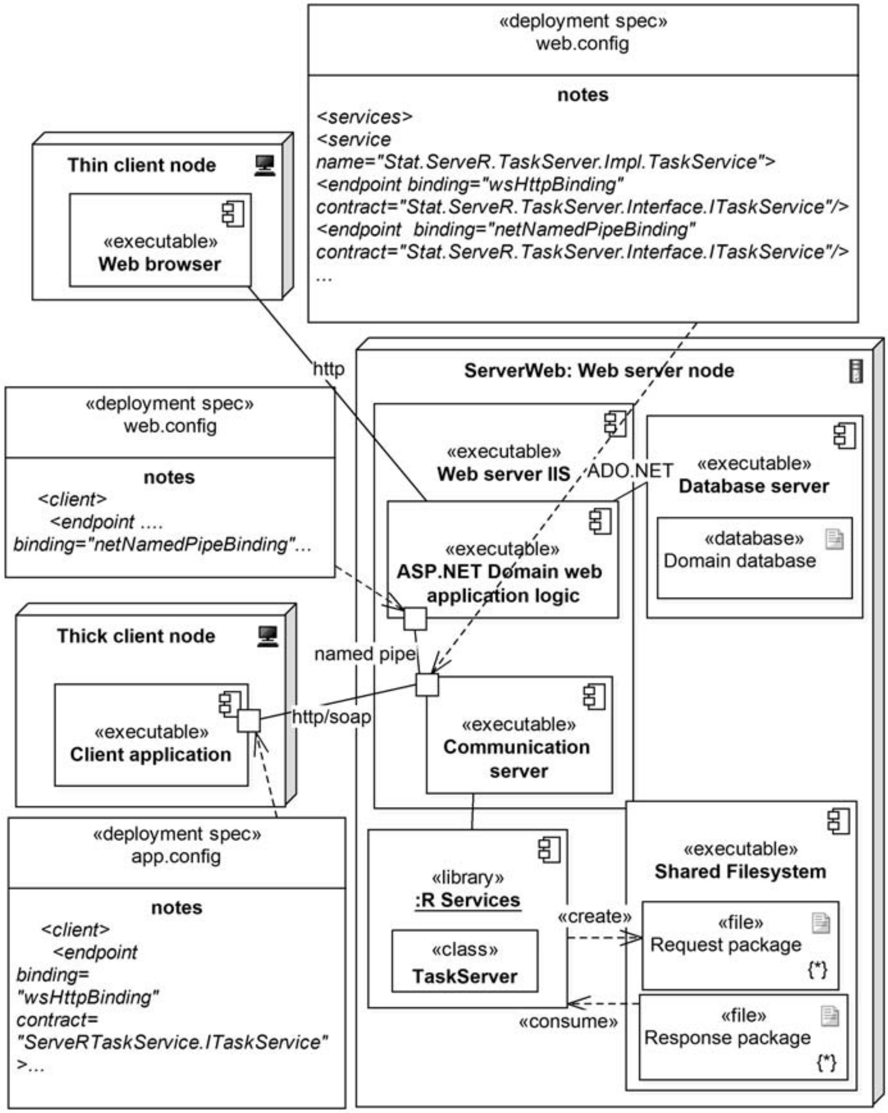



**Fig. 2.** Deployment diagram for client application and communication server

graphs) is similarly delivered to *CliApp*. Every request and response packages are stored in file system of *CommSvr* machine). *CliApp* checks whether response is available on server using pooling mechanism.

The *CliApp* source codes of proxy classes (e.g. in C#) for connectivity with CommSvr can be easily generated automatically from service description WSDL exposed by *CommSvr*. The *CommSrv* machine additionally hosts a domain ASP.NET application with database (the medical system), requests and response package files, MS queue service (Fig. 3).

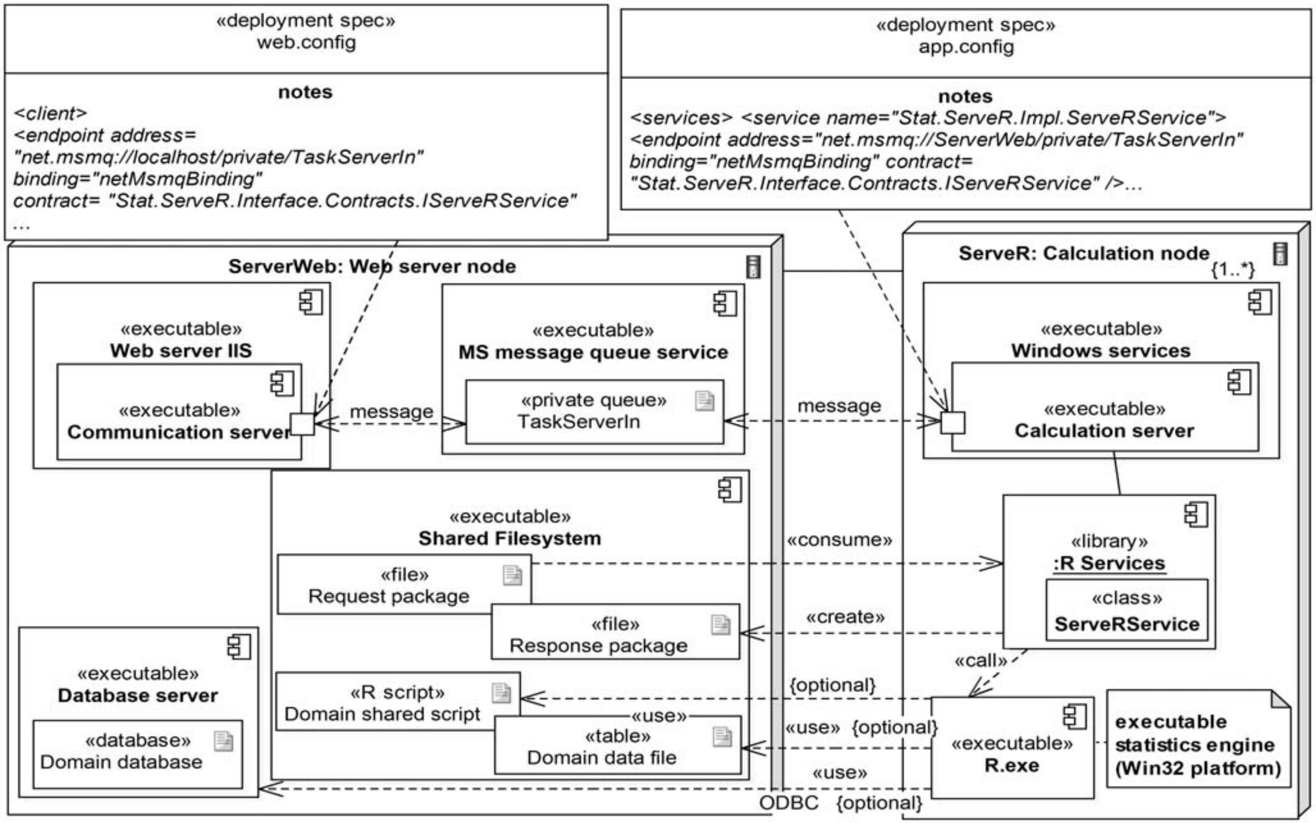



**Fig. 3.** Deployment diagram for communication server and calculation one

## 5 Communication Server and Calculation One

The deployment diagram for *CommSvr* and *CalcSvr* is shown on Fig. 3. *CommSvr* communicates with *CalcSvr* using messaging system (in deployment specification, in client element, binding attribute is set to "netMsmqBinding"). Message queuing prevents losing the requests. *CalcSvr* is hosted by Windows services mechanism. *CalcSrv* takes an information about a request form message queue (in endpoint element, binding attribute is set to "netMsmqBinding") and the request package from the file system shared by *CommSvr* host.

*CommSvr* host additionally exposes for *CalcSvr* common R-scripts, input data files and domain database (accessed by ODBC interface form R scripts [10]). *CalcSrv* uses R.exe statistical engine for executing task for given request package. *System.Diagnostics.Process* class from .NET Framework is used to call R.exe module (To achieve configuration simplicity and stability of *CalcSvrs* installed on potentially many machines, the solution based on calling R via COM interface [7] wasn't applied.). *CalcSrv* holds pool of R.exe module process instances, so not every request causes calling a new process.

*CalcSvr* doesn't produce any final file on *CalcSvr* host. After computing the request task, a response package file is placed in *CommSvr* shared file system.

## 6 On-Line Mode – Remote R Session

Beside the described before batch mode, the system enables functionality of remote R session. In this on-line mode a user can run ad'hoc created simple scripts or even single R commands using remote R console. This mode is useful at interactive R script developing or debugging phases.

In batch mode *ServeR* has three layer architecture. In on-line mode there are only two software layers. *CliApp* and *CommSvr* take part in processing on-line statistics tasks – the calculation services are hosted by *CommSvr*. *CalcSvr*'s logic in embedded into *CommSvr* and exposed as a WCF service, so this layer is denoted as *CommCalcSvr*.

Communication between *CliApp* and *CommCalcSvr* is bi-directional and bases on callback mechanism (binding attribute is set to wsDualHttpBinding in WCF configuration file for both *CliApp* and *CommCalcSvr*). In on-line mode this communication method is more flexible for short running tasks then the pooling mechanism used in batch mode.

*CliApp* can send a request to *CommCalcSvr* and exposes two callback methods (*HandleResponse*, *HandleOutputFile*) which are available for *CommCalcSvr*. *HandleResponse* takes and shows script output (content of stdout after performing request). *HandleOutputFile* takes names and contents of output files (data file, graph) produced by the request execution.

Executing a request at *CommCalcSvr* side relays on passing R code to standard input of instance of *Process* class for R.exe module. The *Process* object can raise the .NET event (*OutputDataReceived*) when R.exe writes on standard output. While handling this event, the remote callback method *HandleResponse* is invoked on *CliApp*, so a user can see what R-scipts write on remote R console.

The process is running in some working directory, where an output file can appear during R commands executing. Using *FileSystemWacher* component makes *CommCalcSvr* sensitive on changes in the process working directory. When a new file is created in that directory, an event is raised (in *FileSystemWatcher*) and in its handler a remote callback method *HandleOutputFile* is invoked on *CliApp*. So output files are automatically delivered to *CliApp* and the user is notified about that.

This WCF-based mechanisms used for on-line mode implementation enables ergonomic interactive remote R session.

## 7 Conclusions

In today‘s world every system should be SOA-ready what means that it should be made of several components (that offer some business services) distributed over network and accessed by other applications. SOA concept helps creating scalable and flexible architecture. Discussed system was designed with four SOA tenet [8] in mind.

Proposed solution shows how .NET infrastructure (WCF and core framework components) can help in exposing command line tools and other standalone applications as services accessible over the network. It uses well known network protocols (HTTP, TCP/IP) and messaging standards (SOAP, WS-*), which makes it easy to access those services from other systems working on different platforms (interoperability feature).

Thanks to WCF all communication and network settings such as transport protocols used, service addresses, messages and transport channels security can be set in configuration file without writing any line of code. This provides maximum flexibility when configuring solution to meet often changeable business needs.

Presented solution was integrated and deployed with existing medical applications which require some statistical computation, but can be used with any other domain-specific systems.

In business scenario computing statistics can be very complicated and takes long time to complete. When number of client application grows it may be necessary to increase efficient system capability. Proposed architecture allows to achieve this by adding computation servers spanned over network nodes. (Also increasing number of web servers that host communication server (see Fig. 2) can improve response time of whole system).

## References

1. R Development Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria (2008) ISBN 3-900051-07-0
2. The R Project for Statistical Computing Page, `http://www.r-project.org`

3. Chemibiogrid project. Statistics (R Web Services) Page, http://www.chembiogrid.org/projects/proj_statistics.html
4. General Rweb Interface Page, http://bayes.math.montana.edu/Rweb
5. Newton, R., Wernisch, L.: Rwui: A Web Application to Create User Friendly Web Interfaces for R Scripts. R News Journal (2008)
6. Create a user friendly web interface for an R script Page, http://rwui.cryst.bbk.ac.uk
7. R COM Connectivity Page, http://rcom.univie.ac.at
8. Peiris, C., Mulder, D., Cicoria, S., Bahree, A., Pathak, N.: Pro WCF: Practical Microsoft SOA Implementation. Springer, New York (2007)
9. Microsoft Developer Network Page, http://msdn.microsoft.com
10. Lapsey, M., Riplay, B.D.: The RODBC Package. ODBC Database Access Page, http://cran.r-project.org/web/packages/RODBC/RODBC.pdf

# Modelling of Multi-tier Internet Applications with the Use of BCMP Queueing Networks and Simulation Model in Simulink

Andrzej Imiełowski

The State School of Higher Vocational Education in Jaroslaw,
Information Science Institute
aimielow@pwszjar.edu.pl

**Abstract.** Internet applications such as Internet stores, reservation systems, on-line banking and a number of other solutions, mainly from the domain of e-business are becoming in recent years more and more important elements of our everyday life. Many Internet applications employ a multitier architecture, which consists of three tiers: a frontend Web tier, a middle-tier that implements core application functionality and a backend database tier. In association with it, also architecture of cluster of servers becomes multitier. Many examples of modeling individual tiers are known in literature. In this article, we describe a model of 3-tier Internet application based on queuing networks consistent with the BCMP model. The analytic model and connected with it calculations of steady-state and performance parameters are described, in the further section we describe the simulation model built in the Matlab Simulink environment.

## 1 Introduction

The last years of the 20th century have brought a rapid raise of the Internet's popularity, which is followed by a development of its new appliances. Applications such as Internet stores, reservation systems, on-line banking and a number of other solutions mainly from a sphere of e-business are becoming more and more important elements of our everyday life, what is followed by a colossal growth of amount of users of these solutions. New appliances, especially from a business sphere, forced the development of new solutions for construction of applications and infrastucture essential for their effective exploitation. Development and better availability of the broadband Internet access is an additional factor that is increasing an acceptable for end-users reaction time and generally speaking exacting an increase of performance of applications.

Present-day Internet applications are complex systems both from the side of the programming architecture and the applied hardware solutions. Typical e-commerce applications are built in the 3-tier architecture: a frontend Web tier, which is responsible for processing HTTP queries and presenting results, a middle-tier, in which a functionality of application is implemented and a backend

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 200–209, 2009.

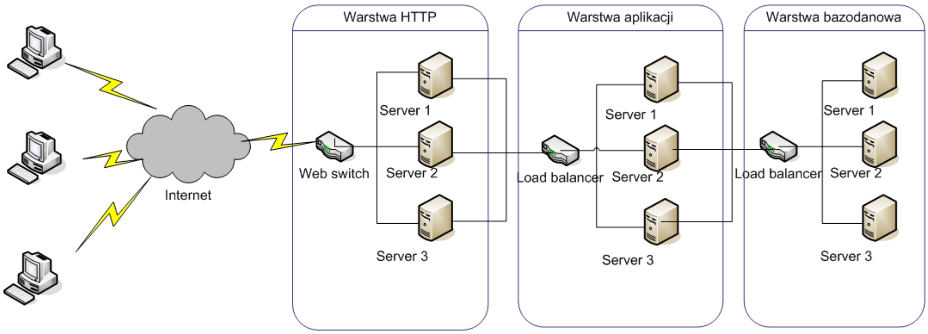



**Fig. 1.** Cluster schema

database tier, which is responsible for storing the data that is being processed by an application.

Various programming solutions are applied in all tiers. Typical and the most often used example of software in the first tier is Apache HTTP Server. According to the research by Netcraft (www.netcraft.com) that provides statistics of Internet services in November 2008, 50.34% of active Internet services were operated by software of various versions of Apache. Second by popularity is Microsoft Internet Information Server software, which is 34.49% according to the same research. Application tier is traditionally dominated by solutions based on PHP. This language was designed in the mid 90s mainly to handle dynamic Internet servers, especially often used with Apache server. Solutions based on Java are gaining more and more popularity. In connection with the huge increase of popularity of Internet solutions by Microsoft, also application solutions of that corporation such as ASP and .NET gain importance. In database tier practically all more important database systems (from open MySQL to commercial solutions such as Oracle or SQL Server) are being used. It should be noted, that currently, in spite of a downward tendency, the most often used in construction of Internet servers solution is a combination of HTTP Apache server, PHP and database server MySQL.

Hardware solutions are being developed to follow a development of software ones. A structure compatible with tier structure of application is used in advanced server systems intended for handling 3-tier internet applications. In case of solutions demanding scalability and big performance, every tier of an application has corresponding tier in a cluster of servers. The schema of such solution is presented in Fig. 1. A model of this solution is considered in the further part of this article.

## 2 Related Works

In recent years, many propositions of modelling multitiered applications appeared and majority of them is based on analysis of queueing networks.

Examples of proposed modeling methods can be found in [2,6,7,9]. Models described in above-cited articles differ in both methods of modeling particular elements of systems and employed methods of calculation. Different solutions of queueing networks are applied in these models. Most often we deal with closed queueing networks and servers like M/M/1, M/M/n (particularly for modeling WWW servers) or M/M/1/PS. Considerable simplifications are employed in some of described models, especially considering order of processing requests in the system. Generally, the mean value analysis (MVA) algorithm is applied to determine performance parameters. The most complex and composite model is presented in [10]. This model, however, does not take dispatchers into consideration. Division of requests for different classes in view of priorities of system's clients is taken into account in the above-mentioned models. This division of clients is described most thoroughly in [8]. In model proposed in that article, different division of clients is employed on the basis of method of processing requests by the system, which will be described in further part of this article.

# 3 Analitycal Model

## 3.1 Handling Requests in a 3-Tier Application

The schema of processing requests in a 3-tier application is presented in Fig. 2. According to this schema requests are divided into 3 classes depending on the way of handling them in a system. Class 1 – requests that are processed only in a HTTP tier, class 2 – requests that demand handling in an application tier without using a database tier and class 3 containing requests that demand handling in all 3 tiers of a system.

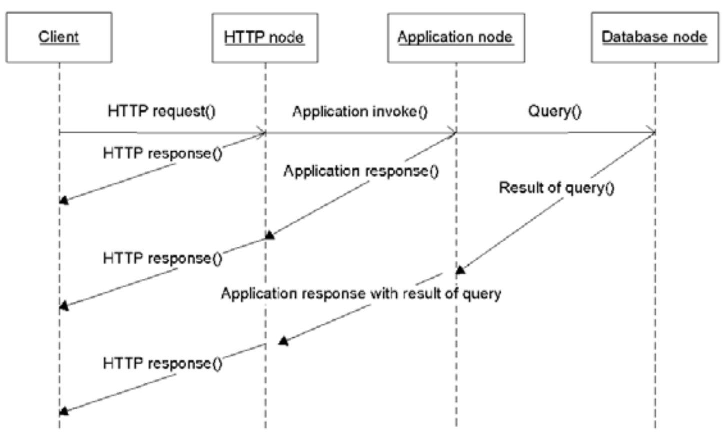



**Fig. 2.** Request handling schema

### 3.2 Model's Description

The analitycal model based on a queueing network compatible with a BCMP (Basket, Chandy, Muntz, Palacios) [1] model is presented in depth in [11]. A schema of the queuing network exploited in the model in presented in Fig. 3.

M/M/1/FIFO type nodes are proposed to be applied as a model of dispatchers, nodes with a distribution of service rates compatible with Cox distribution (–/C/1/PS) is proposed to be applied for the description of servers in individual tiers, particularly Cox distribution second degree is proposed to be employed. We assume that the system is supplied with requests flowing in Poisson stream with constant parameter independent of an amount of clients. Processing times in individual nodes are independent of the amount of clients and the request class. Three classes of client's requests are provided in the model as described in Sect 3.2. In addition, we introduced two auxiliary request classes intended for the description of return route of the request. For example: for the class 3 requests (requiring processing in the database tier) we introduced an additional class describing route of the request from the database server to the client. Likewise, for class 2 requests (processed by an application tier) we introduced auxiliary class describing route of the request from the application through the WWW server to the client. This model $S$ can be described with the following formula:

$$S = (Q_0, (\mu_{i0}, m_i, \mu_{i01}, \mu_{i02}, A_i),\ i = 1..3) \tag{1}$$

where:

$Q_0$ – a routing matrix,
$i$ – a number of the tier,
$\mu_{i0}$ – an intensity of processing time in the $i$-th tier,
$m_i$ – a quantity of servers in the $i$-th tier,
$\boldsymbol{\mu_{i01}}$ **,** $\mu_{i02}$**,** $\mathbf{A_i}$ – parameters of Cox distribution for the service rate for servers in the $i$-th tier.

$Q_0$ matrix is the recording of request processing described in Sect. 3.2.

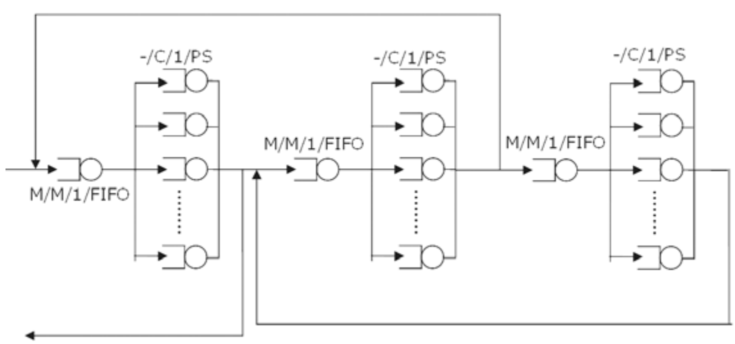



**Fig. 3.** Schema of queue network in model

Request stream $\lambda$ contains three request classes divided in accordance with the probability respectively $p_1, p_2, p_3$. In the model we assumed that dispatchers work according to the Round Robin algorithm, in connection with which the probabilities of passing between a dispatcher and servers in $i$-th tier will be inversely proportional to the amount of servers ($m_i$) in the tier.

### 3.3 Determining Parametres of Model

**Determinig System Steady-state Probability.** Derivation of the formula for calculation of the steady-state probability of described above model is presented in [11], below we present the final formula:

$$p(n) = \lambda^{\bar{n}} \prod_{i=1..2,\, j=0..m_i} \bar{n}_{ij}! \cdot \prod_{k \epsilon R} p_k^{2\bar{n}^{(k)}} \cdot \\ \cdot \left( \prod_{i=1..3,\, j=0..m_i} (1-\rho_{ij}) \cdot \prod_{i=1..2,\, j=0..m_i,\, k\epsilon R} \bar{n}_{ij}^{(k)} \cdot \prod_{i\epsilon W} \mu_i^{\bar{n}_i} \right)^{-1} \tag{2}$$

where:

$n$ – a vector describing a state of all servers
$\bar{n}$ – a number of requests in all servers
$\bar{n}_{ij}^{(k)}$ – a number of requests of $k$-th class in $i$-th tier on $j$-th server
$R$ – a set of request classes
$W$ – a set of tiers
$\rho_{ij}$ – a utilization of $j$-th server in $i$-th tier

In practice, another method of determining steady-state probability will be employed. Equation 2 is too detailed formula, generally determining probability of appearing of a particular state without taking distribution of request classes into consideration is sufficient. For this purpose we can employ the following (known from studies [3,4] on BCMP networks) formula:

$$p(n) = \prod_{i=1..3,\, j=0..m_i} (1-\varrho_{ij}) \cdot \varrho_{ij}^{n_{ij}} \tag{3}$$

Calculating total probability of states of all nodes until reaching given maximal states or states characteristic for the given system may have a frequent practical use. That outcome can be determined by summing probabilities of all possible states calculated from Eq. 3, that is:

$$p_s(n_{max}) = \sum \left( \prod_{i=1..3,\, j=0..m_i} (1-\varrho_{ij}) \cdot \varrho_{ij}^{n_{ij}} \right) \tag{4}$$

Marking by $n_{ijmax}$ and taking into consideration that servers on particular layers are identical we can eventually write the formula as:

$$p_s(n_{max}) = \prod_{i=1..3,\, j=0..m_i} \left( (1-\varrho_{ij}) \cdot \sum_{n=0}^{n_{max}} \varrho_{ij}^{n} \right) \tag{5}$$

Equation 5 can be helpful in calculating probability that given system in any node will not exceed the state recognized as safe or meeting performance conditions given in Service Level Agreement.

**Determining Model Performance Parameters.** The resultant equations describing average system's answer times for particular request classes are presented below:

– for class 1 (request requiring processing only in Web tier):

$$T^{(1)} = \frac{1}{\mu_1 - \lambda\left(p_1 + 2p_2 + 2p_3\right)} + \frac{m_1}{m_1\mu_{11} - \lambda\left(p_1 + 2p_2 + 2p_3\right)} \tag{6}$$

– for class 2 (request requiring processing both In Web tier and in application tier):

$$T^{(2)} = \frac{2}{\mu_1 - \lambda\left(p_1 + 2p_2 + 2p_3\right)} + \frac{2m_1}{m_1\mu_{11} - \lambda\left(p_1 + 2p_2 + 2p_3\right)} + \frac{1}{\mu_2 - \lambda\left(p_2 + 2p_3\right)} + \frac{m_2}{m_2\mu_{21} - \lambda\left(p_2 + 2p_3\right)} \tag{7}$$

– for class 3 (request requiring processing in all tiers of the model):

$$T^{(3)} = \frac{2}{\mu_1 - \lambda\left(p_1 + 2p_2 + 2p_3\right)} + \frac{2m_1}{m_1\mu_{11} - \lambda\left(p_1 + 2p_2 + 2p_3\right)} + \frac{2}{\mu_2 - \lambda\left(p_2 + 2p_3\right)} + \frac{2m_2}{m_2\mu_{21} - \lambda\left(p_2 + 2p_3\right)} + \frac{2m_2}{m_2\mu_{21} - \lambda\left(p_2 + 2p_3\right)} + \frac{1}{\mu_3 - \lambda p_3} + \frac{m_2}{\mu_3 m_{31} - \lambda p_3} \tag{8}$$

All formulas presented in Sect. 3.3 can be employed only when ergodicity conditions are met, i.e. the intensity of request stream in each node has to be smaller than intensity of request handling in that node, what can be written as:

$$\mu_i > \lambda_i \tag{9}$$

## 4 Simulation Model

A simulation model of a cluster sytem that has a structure analogous to the one presented in Sect. 3.1 was prepared for the purpose of the verification of the analytical model presented in Sect 3. Two additional restrictions were introduced in relation to the analitycal model:

– servers (in all tiers of the model) have limited quantity of simultaneously handled requests
– possibility of rejecting a request due to a run-down of server's resources

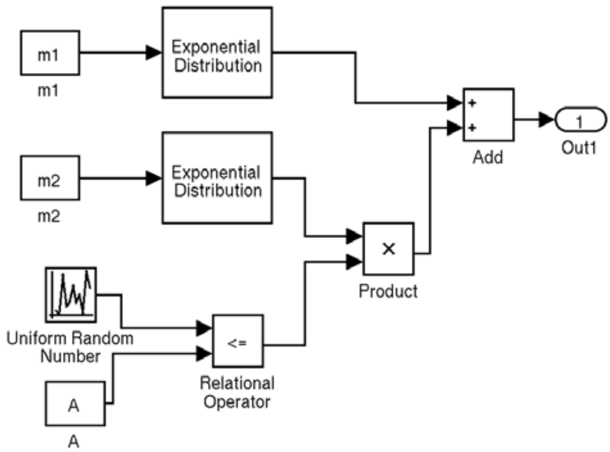



**Fig. 4.** Schema of generation of Cox ditribution's time

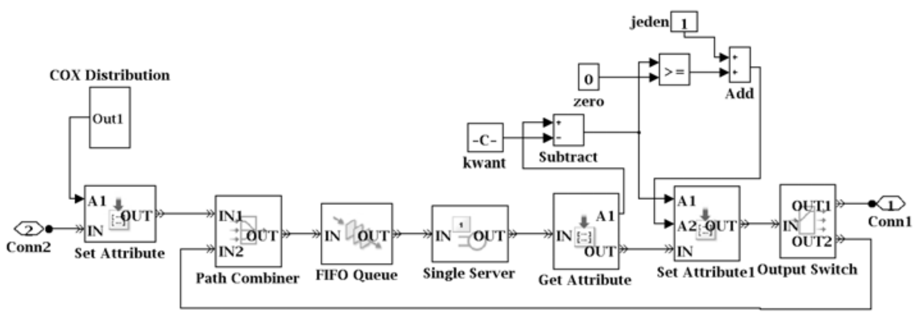



**Fig. 5.** Schema of submodel of server with processor sharing

Simulation model was implemented in Matlab Simulink environment with the use of SimEvent toolbox that is intended for discrete simulations. Standard elements supproted by Simulink were exploited. Preparation of two kinds of submodels was also necessary: the submodel that generates processing times compatible with Cox distribution and the submodel of the server with a distribution of processor's time.

Figure 4 presents a schema of the submodel that generates processing times compatible with Cox distribution. Processing time is generated based on parametres that determine a second degree Cox distribution, where: $\mu_1$, $\mu_2$, A – Cox distribution parameters. Values of the parametres are determined based on the method described in [4].

Figure 5 presents the schema of submodel of single server with processor sharing (PS). Time necessary to process single request is generated according to Cox distribution in the way described above and stored in attribute of request.

Server processes requests in a constant quantum of time. Requests are being buffered in a FIFO queue. After every request's passing of server the attribute, in which time remaining to end handling, is decreased by the quantum of time. A request leaves submodel if value of the atribute falls below zero, otherwise it returns to the queue and waits for allocation of a server.

## 5 Simulation Experiment

With the use of the analytical model described in Sect. 3, we calculated performance parametres for the system with the parametres given in the Table 1. For calculations we assumed the following division of arriving request stream $p1 = 0.2$, $p2 = 0.2$ and $p3 = 0.6$. Input data for calculations is presented in the Table 1:

**Table 1.** Parameters of test system

| | $\mu$ | $\mu_i$ | $m_i$ |
|---|---|---|---|
| tier 1 | 5000 | 500 | 5 |
| tier 2 | 5000 | 650 | 3 |
| tier 3 | 5000 | 300 | 3 |

So far, we conducted preliminary simulations, the comparative results are presented in Fig. 6 and Fig. 7.

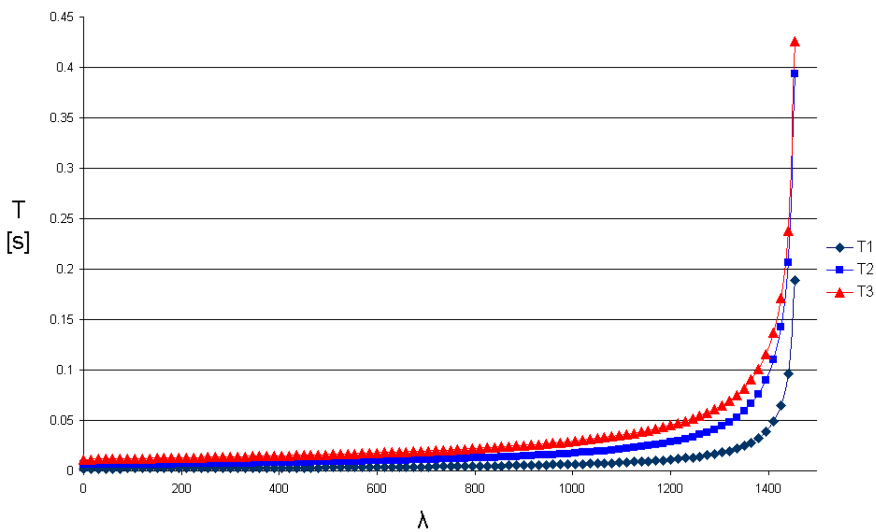



**Fig. 6.** Time of response derived from analytic model

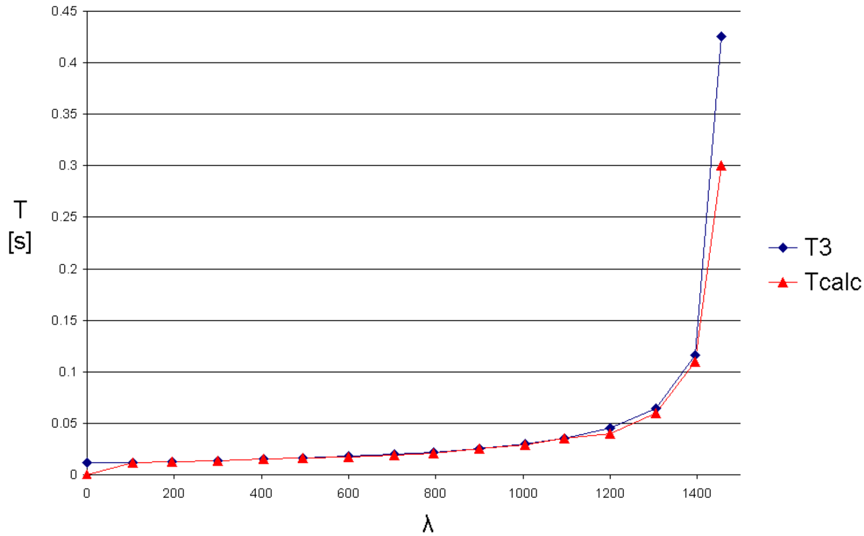



**Fig. 7.** Time of response calculated from model and from simulation

## 6 Conclusion and Future Work

Initial results of simulation indicate that analytical model can be useful to determine performance parametres of cluster systems that handle multitier applications. Obviously, for the full confirmation of the usefulness of the model it is essential to conduct more experiments with different system configurations. Only a full anlysis of the results for a wide range of tests would permit us to determine the real usefulness of the analytical model.

## References


1. Baskett, F., Chandy, K., Muntz, R., Palacios, F.: Open, Closed, and Mixed Networks of Queues with Different Classes of Customers. Journal of the ACM 22(2), 248–260 (1975)
2. Bhulai, S., Sivasubramanian, S., Van der Mei, R., Van Steen, M.: Modeling and Pre-dicting End-to-End Response Times in Multi-tier Internet Applications. In: Managing Traffic Performance in Converged Networks, pp. 519–532. Springer, Heidelberg (2007)
3. Bolch, G., Greiner, S., de Meer, H., Trivedi, K.S.: Queueing Networks and Markov Chains: Modeling and Performance Evaluation With Computer Science Applications. Wiley Interscience, Hoboken (2006)
4. Czachórski, T.: Modele kolejkowe w ocenie efektywności sieci i systemów komputerowych, Gliwice, Pracownia Komputerowa Jacka Skalmierskiego, 109–126 (1999)
5. Gelenbe, E., Pujolle, G.: Introduction to Queueing Networks. Other Wiley Editorial Office's, New York (1998)

6. Liu, X., Heo, J., Sha, L.: Modeling 3-tiered Web applications. In: Proceedings of the 13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, pp. 307–310 (2005)
7. Menascé, D.A., Almeida, V.A.F., Dowdy, L.W.: Performance by Design: Computer Capacity Planning by Example, pp. 178–195. Prentice Hall PTR, Englewood Cliffs (2004)
8. Menascé, D.A., Almeida, V.A.F.: Scaling for E-Business: Technologies, Models, Performance, and Capacity Planning. Prentice Hall PTR, Englewood Cliffs (2000)
9. Urgaonkar, B., Shenoy, P., Chandra, A., Goyal, P.: Dynamic Provisioning of Multi-tier Internet Applications. In: Proceedings of the Second International Conference on Autonomic Computing (ICAC 2005), Washington, pp. 217–228. IEEE Computer Society Press, Los Alamitos (2005)
10. Urgaonkar, B., Pacifici, G., Shenoy, P., Spreitzer, M., Tantawi, A.: Analytic Modeling of Multitier Internet Applications. ACM Transactions on the Web 1(1), Article No. 2 (2007)
11. Imiełowski, A.: Analytical Model of Multitiered Internet Applications with the Use of BCMP Queueing Networks, Information Systems Architecture and Technology. In: Web Information Systems: Models, Concepts and Challenges, Wrocław, Oficyna Wydaw. PWroc., pp. 113–124 (2008)

# Disaster's Impact on Internet Performance – Case Study

Tomasz Bilski

Poznań University of Technology
tomasz.bilski@put.poznan.pl

**Abstract.** The paper presents how disasters change Internet performance. In the first section, transmission performance measures are defined with description of expected values for such applications as VoIP and Video. Next section deals with performance monitoring tools and services. The main part of the paper is dedicated to analysis of exemplary damage and its impact on Internet performance. Meditteranean accident (in January 2008) with submarine cables breakage is described. The accident had serious impact on Internet performance. In the last part of the paper we will suggest some recommendations for future networks.

## 1 Introduction

Long distance connections are based on telecommunication cables and satellite tranceivers. Satellite transmission parameters are relatively poor, with significant delay and small bandwidth (signal propagation via geostationary satellite takes 260 ms [1]). So long distance communication is based mainly on cables. The cables are susceptible to breakages caused by earthquakes, storms and other sources[1]. Many Internet applications require high level of reliability: VoIP, videoconferences, interactive applications. Reliability and performance is disrupted by intentional and unintentional factors: malicious software, spam, hackers, disasters. First 3 factors are widespread. Damages imposed are usually restricted to single services and are short-lived. Disasters are uncommon but their impact is long-lived. They disrupt Internet services, telephone calls and ATM (Automated Teller Machine) transactions. Internet performance is an important research area. Many studies are based on models, simulations and test networks. Such research is valuable as long as the assumptions, models and testbeds are accurate. In this paper we are analyzing authentic data taken from genuine network in real time. An accident presented in the paper is well documented. IEPM (Internet End-to-end Performance Measurement) at SLAC (Stanford Linear Accelerator Center) preserves data (RTT, throughput, jitter) on net performance during the accident [3]. RIPE [5] (Réseaux IP Européens) and Renesys [6] analyzed IP route changes. Study presented here tries to reveal more compound analysis with some recommendations for network improvement.

[1] Satellites are also vulnerable, they may move away from orbit or may collide (on Feb 11 2009 Iridium Satellite collided with Russian Cosmos 2251 satellite, an incident resulted in limited disruptions of Iridium service).

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 210–217, 2009.

## 2 Internet Performance Measures

Transmission performance is measured with a use of some quantitative indicators. Most commonly applied are: throughput, delay, jitter and loss packet ratio. The values of the parameters are related to many, static and dynamic, factors: communication medium and infrastructure, communication protocol (e.g. UDP throughput is higher than TCP), distance between sender and receiver (measured in hops, but also in kilometers), current network load.

### 2.1 Throughput

Throughput is defined as the average rate of successful data delivery. Measured in bits/s or in packets/s. Throughput is a fraction of channel bandwidth, which is related to link layer technology (e.g. Ethernet, ATM). Throughput level is an important factor in audio and video real time communication. For example single VoIP channel needs about 90 kbit/s for G.711 codec.

### 2.2 Delay and Jitter

Delay in IP networks is a time between sending and receiving IP datagram. Delay is very important in interactive, real-time applications such as VoIP. Large value of delay makes it impossible to satisfy VoIP users (Table 1). It must be noted that, mouth to ear delay is a total delay. It consists of coder/decoder delays, jitter buffer delay, packetization delay. IP datagram delay is only a fraction of it. So the required datagram delays are much lower (in the worst case lower of about 150 ms) than mouth to ear delays. Packet delay is a function of distance (measured in kilometers) and number of routers in the communication channel. In the IP network environment RTT defined as the sum of the delays for two transmission directions is used.

IP transmission exhibits variable delay (jitter) in packet delivery time. Jitter is the absolute value of the difference between the forwarding delay of two consecutive received packets belonging to the same stream [2]. Small jitter may be removed prior to replaying speech with a use of jitter buffer which stores incoming packets and sends them in a more constant stream. The buffer introduces additional time to mouth to ear delay. So the buffer compensates jitter not greater than 50 ms. Large jitter introduces significant degradation of the call.

**Table 1.** Effects of the absolute delay according to the E-model [1]

| Mouth to ear delay [ms] | Quality of service |
|---|---|
| Less than 200 | users very satisfied |
| 200–300 | users satisfied |
| 300–400 | some users dissatisfied |
| 400–550 | many users dissatisfied |
| More than 550 | nearly all users dissatisfied |

### 2.3 Packet Loss Rate

IP network does not assure that each packet is delivered to the receiver. Some packets are lost, usually due to router congestions. If router input queue is full then next received packet is dropped[2]. Packet loss rate is a good indicator of link/network quality especially in the case of interactive applications (Table 2). High packet loss rate makes the applications unusable (e.g. video becomes irritating, VoIP users become unable to communicate). It must be noted that for a given packet loss rate the service quality is a function of codec. Higher codec compression means more disturbing effect of lost packets.

**Table 2.** Packet loss rate and interactive application quality

| Percentage of lost packets | Interactive application quality |
|---|---|
| 0–1 | good |
| 1–2.5 | acceptable |
| 2.5–5 | poor |
| 5–12 | very poor |
| more than 12 | bad |

## 3 Internet Monitoring Tools and Services

Network performance may be checked with a use of dedicated protocols such as SNMP, RTCP and ICMP. Especially ICMP Echo mechanism is helpful. It allows to send packets to a node and have them echoed back, so the sender may calculate e.g. RTT and jitter. Nowadays ICMP usually is pre-installed on almost all platforms. The server (the echo responder) runs at a high priority (e.g. in the kernel on Unix) and is more likely to provide a good measure of performance than a user application. Ping tool is based on ICMP Echo messages.

There are many services for Internet performance measurement. Some of them are integrated with databases of past measurements[3]. The example is PingER, a service monitoring performance of Internet links, developed at SLAC and operating since 1998. Monitoring is based on more than 300 distributed hosts. The system sends periodically pings for each tested connection. The measurement results are written to database[4] [3]. Routing analysis is done by RIS (Routing Information Service), managed by RIPE NCC. It collects (in near real time) BGP routing information messages from 600 peers at 16 exchange points mainly in Europe and US. Results are stored in a database for further processing. Three times a day, the route collectors take snapshots of their respective Routing Information Bases[5].

[2] In some queue management algorithms packets are dropped even if queue is not full.
[3] http://www.slac.stanford.edu/comp/net/wan-mon/netmon.html
[4] http://www-wanmon.slac.stanford.edu/cgi-wrap/pingtable.pl
[5] http://www.ripe.net/projects/ris/

# 4 Disaster Analysis

## 4.1 Introduction

Cable faults are relatively infrequent (in 2003 annual fault rate was at the level of 1 fault per 10000 km of cable [7]). In the case of faults channel redundancy plays an important role. Cable breaks in the Atlantic happen repeatedly (more than 50 cable repairs are yearly in the Atlantic) but due to the high level of redundancy they are almost invisible to end users. On the other hand disasters on Mediterranean inflict big impact on Internet services. Submarine cables are prone to being affected by earthquakes, storms, fishing and anchors. 70% of faults are attributed to human activity with fishing as the major cause [7]. Usually earthquake extent of the damage is much greater, the cable may suffer several breaks. The severed ends could be buried by deep-sea landslides or washed kilometers from their positions. It may take days to just find the cable[6].

## 4.2 Recent Accidents

In June 2005 SMW 3 cable had been cut off Karachi. Pakistan lost all terrestrial Internet connectivity. The outage of services lasted 12 days. On December 26 2006 a 7.1-magnitude earthquake south of Taiwan knocked 7 submarine cables out of service, impairing communications from North America to China, Taiwan, Japan, Korea as well as inside North and Southeast Asia. Cables accounting for 90% of telecommunications capacity of the region had been broken[7]. It took 49 days to repair all the cables. On January 30 2008 started a series of accidents with Mediterranean cables damaged. The case will be analyzed more thoroughly in the next subsection. On December 19 2008 five submarine cables (including SMW 3, SMW 4, FLAG, Seabone) had been cut near Sicily due to 5.3 magnitude quake in the central Mediterranean. The cut disrupted Internet and telephone services in the region and in parts of the Middle East and South Asia [4].

## 4.3 January 2008 Mediterranean Accident

Mediterranean accident is an interesting research subject from many points of view. It consisted of several events. It occured in the area of high traffic and relatively small connection redundancy. It demonstrated several problems with current Internet infrastructure.

**Accident Timetable.** The accident was a series of events. Cables from Europe to Middle East and Asia were affected: Jan 30 about 4:30 (UTC) SMW 4 cable near Alexandria was damaged, Jan 30 about 8:00 (UTC) FLAG cable near Alexandria was damaged (the two cables carry about 70% of the traffic between Europe and the Middle East), Feb 1 about 6:00 (UTC) FALCON cable near Dubai was damaged, Feb 8 SMW 4 cable was repaired, Feb 9 FLAG cable was

[6] Communication problems in the area of earthquake are especially serious issue.

[7] http://www.iht.com/articles/2006/12/28/business/connect.php

repaired, Feb 10 FALCON cable was repaired [5]. It is assumed that the cables were damaged by anchors.

In the next part we analyze the impact on IP routes and performance. The accident imposed service degradation on many communication channels. Two representative connections from Europe to India (cern.ch to ernet.in (1) and ictp.it to cdacmumbai.in (2)) and two connections from US to India[8] (ascr.doe.gov to ernet.in (1) and ascr.doe.gov to cdacmumbai.in (2)) are explored. The analysis is based on IEPM PingER and RIPE RIS databases.

**Impact on IP Routes.** It may be expected that submarine cable cut causes changes to BGP routes. Three general types of changes are possible: immediate loss of connection between some networks after the failure, much smaller number of available AS (Autonomous System) paths and rerouting with a use of backup paths (if they exist). Of course these backup paths are longer and offer poorer performance[9]. A good indication of the impact is the number of IP network prefixes, that are announced in BGP messages. If prefix to a given network is not announced to routing peers then the network is not reachable. In some countries (Egypt, Kuwait, Sudan), immediately after the failure, more than 30% of the prefixes were removed from BGP announcements. In the effect many networks disappeared from the routing tables. The total number of AS paths decreased. At the same time the number of changes in AS paths rapidly increased in the region. During normal operation not more than 10% of paths change daily. On Jan 30 more than 60% of paths had been changed in the region. This higher than normal and fluctuated (10–30%) paths change percentage persisted to Feb 18. Next alteration is visible in the average AS path length (the number of different ASes between two distant networks). Around the time period of the cuts it slightly increased from 5.5 to about 6.

Most disrupted Europe-Asia connections were rerouted through the SMW 3 cable or fibres taking the way around the globe (Europe-US-Asia). Due to the limited bandwidth and traffic increase on these new routes it was difficult to converge routing tables on alternate topologies [5]. Here we see the consequences of using distance-vector algorithm for BGP routing optimization. Furthermore this traffic rerouting had significant impact not only on rerouted connections but also on some of the US to Asia connections (see next subsections). IP routes modifications resulted in significant changes of RTT, throughput and packet loss ratio. All IP routing parameters returned fully to their values from before accident only several days after the time in which all repairs were completed.

**RTT and Jitter.** Average monthly RTT in Dec 2007, for both Europe to India connections, was between 170 and 185 ms. Connections between Europe and India had been significantly affected: average RTT increased on Jan 30 to above 600 ms. Changes were inflicted by traffic rerouting through US networks which

[8] India is an IT outsourcing centre, so the connections to India's networks are particularly important for US and Europe businesses.

[9] Rerouting may be done also at link layer. In this case IP route after the failure does not change but performance indicators (throughput, delay) become worse.

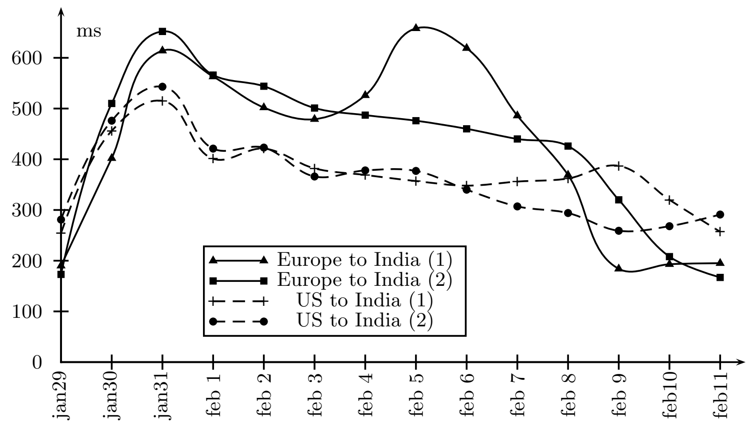



**Fig. 1.** RTT from Europe to India and from US to India

started to carry additional traffic. In consequence connections between US and India had been affected too: average RTT (US to India) increased on Jan 30 and 31 from below 300 (in Dec 2007) to above 500 ms (Fig. 1). RTT increased due to longer routes, more routers and longer queues in routers, which have to carry additional traffic. Average RTTs started to improve (with some fluctuations) after Jan 31 and returned to their values from before accident after 10–12 days.

Average monthly jitter for Europe to India connections was (in Dec 2007) at the level of 8–9 ms. It slightly increased during first 2 days to 11 ms and sharply increased on Feb 1 to above 60 ms. Next days it started to improve. Average monthly jitter for US to India connections was in Dec 2007 at the level of 7–12 ms. It increased during first 2 days to 12–19 ms. The highest jitter 24–32 ms appeared on Feb 8 and 9, at the time of cable fix.

**Throughput.** Average monthly throughput (TCP connections with 1000 bytes per packet) for Europe to India connections was in December 2007 at the level of hundreds kbit/s. It rapidly decreased on Jan 30 to 26–39 kbit/s. Similarly average monthly throughput for US to India connections was in Dec 2007 at the level of hundreds kbit/s. It rapidly decreased on Jan 30 to 31–39 kbit/s. (Fig. 2). Throughput remained below 200 kbit/s up to Feb 9 with temporary improvement (in selected channels) on Feb 5. It may be observed that this Feb 5 throughput anomaly is visible on Europe to India (1) RTT plot.

**Packet Loss.** Average monthly packet loss for Europe to India connections was in Dec 2007 at the level of 0–7%. It increased after cable fault to 23–46%. This level of packet loss means total unavailability of interactive services. Average monthly packet loss for US to India connections was in Dec 2007 at the level of 0–4%. It increased on Jan 30 to 40–44%. (Fig. 3). Packet loss temporarily recovered in next few days. On Feb 3 we observe decrease for Europe to India

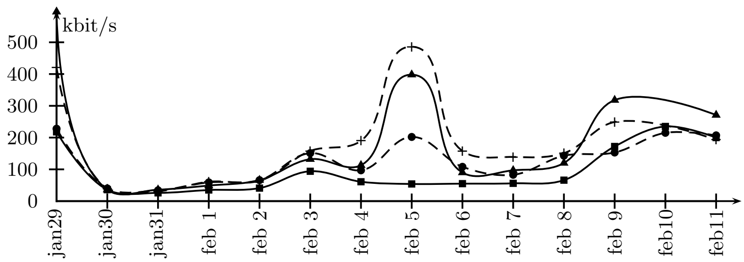



**Fig. 2.** Throughput from Europe to India and from US to India

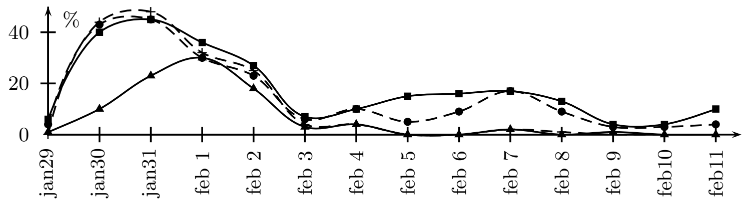



**Fig. 3.** Packet loss rate from Europe to India and from US to India

and US to India connections. In both cases the level not greater than about 10% has been achieved. In the next few days it fluctuated between 0 and about 20%.

## 5 Conclusions

Internet survived the accident. Nevertheless the disaster's impact on its performance was: significant, long-lived and widespread (cable cut near Alexandria degenerated US to Asia connections which do not use the cable). The performance parameters during the accident decreased to unacceptable (for interactive applications) levels. Internet performance have been changing in unpredictable way (it is seen for example on throughput and RTT plots). The effects of submarine cable cut accident are notably different from the effects of hacker attack on Internet server. The analyzed accident was not an exception. Similar accidents in the future should be expected. Generally we are not able to predict time and place of the accidents (earthquakes are hardly predictable).

Many things should be done to improve Internet performance and reliability. Redundancy should be carefully planned at every infrastructure level. For end user it is pointless to use two ISPs if both utilize the same international cable. Of course more cables are needed. Their location should be better planned. Existing cables should be upgraded so that they are operating at a percent of their potential capacities, leaving plenty of room not only for future traffic growth but also for rerouted (in the case of disaster) traffic. Globally used services should be based on data mirroring, caching proxies, CDNs (Content Delivery Networks).

Applications vary widely in their QoS requirements, so they should be better profiled before determining appropriate classification and routing treatment. Both versions of IP are ready for such data classification. Routers should be aware and able to carry differentiated traffic. More efficient implementations of TCP should be integrated with common operating systems. BGP protocol should be upgraded or replaced with completely new one. Routing table convergence time should become important optimization criterion. To save bandwidth real time applications should be based on more efficient codecs. RTP/UDP/IP header compression should be broadly utilized. It must be noted that suggested improvements are not easy to implement but many of the techniques, algorithms, protocols and their implementations are already available.

## References

1. ITU-T Recommendation G.114. One-way transmission time, ITU-T 05/2003
2. Poretsky, S., et al.: Terminology for Benchmarking Network-layer Traffic Control Mechanisms, IETF (2006)
3. Cottrell, L., Matthews, W., Logg, C.: Tutorial on Internet Monitoring and PingER at SLAC, SLAC (2007), http://www.slac.stanford.edu/comp/net/wan-mon/tutorial.html
4. Cottrell, L.: Effects of Mediterranean Fibre Cuts December 2008, SLAC (2009), https://confluence.slac.stanford.edu/display/IEPM/Effects+of+MediterraneanFibre+Cuts+December+2008
5. Wilhelm, R., Buckridge, C. (eds.): Mediterranean Fibre Cable Cut – a RIPE NCC Analysis RIPE (2008), http://www.ripe.net/projects/reports/2008cable-cut/index.html
6. Popescu, A., Premore, B., Zmijewski, E.: Impact of the Middle East Cable Breaks. A Global BGP Perspective, Renesys Corp. (2008), http://www.renesys.com/tech/presentations/pdf/nanog42-lightning.pdf
7. Kordahi, M.E., Shapiro, S.: Worldwide Trends in Submarine Cable System Faults, SubOptic (2004), http://www.suboptic.org/Viewdocument.aspx?id=381

# Creating 3D Web-Based Viewing Services for DICOM Images*

Adam Piorkowski, Lukasz Jajesnica, and Kamil Szostek

Department of Geoinfomatics and Applied Computer Science,
AGH University of Science and Technology, Cracow, Poland
pioro@agh.edu.pl
http://www.agh.edu.pl

**Abstract.** This paper presents the design of Web-based medical image viewing system. It allows easy access to the selected DICOM data. The purpose of this application is to provide 3D results of computer tomography as a projection at any angle. This system contains of modules for data storing, projection creating and image file creating. In the experimental part two common web technologies (Java and ASP) were tested in terms of performance for designed system.

## 1 Introduction

The telemedicine and medical image processing has recently become more and more popular. Its main objective is to provide the results of medical examinations on-line, in the real time. Such applications require the creation of special programs that can share, process and transmit a lot of medical data. A simpler solution is to create a web based application that meets the above requirements. This type of program requires only a web browser on the client side. Disadvantages of this case are requirements for performance of hardware (high performance of servers) and performance of software (server-side computing). This article also shows a review for performance of server-side image processing (Java and ASP).

On-demand server-side image processing was the focus of the work by T. Sakusabe et al. in 2000 [1]. The authors presented their own concept of a system that takes advantage of the user's interaction in script languages and processes images on the server's side. They selected and tested the following technologies of applications: ISAPI (currently obsolete, replaced by ASP) and CGI. The method of compression images (PNG vs. JPG) was also tested. The main goal of this work was the performance test for various network connections, bandwidths and client loads. Our considerations relate in some parts to their work.

A new approach to create and manage image-based electronics patient records from actual patient records was a topic of work by Z. Jianguo et al. [2]. There

* This work was financed by the AGH – University of Science and Technology, Faculty of Geology, Geophysics and Environmental Protection as a part of statutory project number 11.11.140.561.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 218–224, 2009.

is also presented a way to use Web technology and DICOM standard (Digital Imaging and Communications in Medicine) to build an open architecture for collaborative medical applications. The focus is on transmission security of patients' data in the Internet/intranet.

Paul Ancuta presented a sample 3D modeling of human bone in the paper [3]. To make this visualization technique particularly useful, the author considered a isosurface method. The image is created from successive computer tomography (CT) images. The processing is done by standalone software here.

A multidimensional image navigation and display software was a topic of paper by Antoine Rosset et al. [4]. The authors presented their own standalone solution to display and interpret large sets of multidimensional and multimodality images such as combined PET-CT studies. To optimize display they used 3D graphic capabilities of the OpenGL graphic standard.

Joseph Fernandez-Bayo et al. showed an application for distributing medical images on the Internet [5]. They implemented a server (or cluster of servers) for picture archiving and communication systems (PACS) that served sets of DICOM images over network (under control of Windows NT). They also implemented image viewer (using Java) on the client side. The classic PACS system, which is an open source, was also presented in this article [6].

## 2 DICOM Images

The results of some medical methods of examination are sets of images. This is the case of e.g. computer tomography. The CT machine creates a cross-section in the form of slices with a given distance away. As a result we get a set of images, which are stored in a special lossless format – DICOM. Although most of DICOM images use grayscale palette, the resolution can be higher (e.g. 12 or 16 bits) than standard display systems (8 bits for CRT, LCD). It is therefore necessary to convert the data to view the images properly.

## 3 Creating a Projection of 3D Image

To create a projection of 3D image a block of data is created (Fig. 1). Each next slide of CT result is taken as the subsequent surface for Z value. Values (X,Y) are the pixels of surface. The client can ask a server for any slice in block. It can

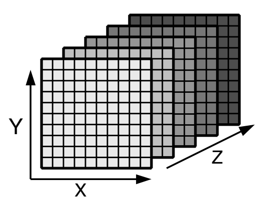



**Fig. 1.** A block of DICOM images

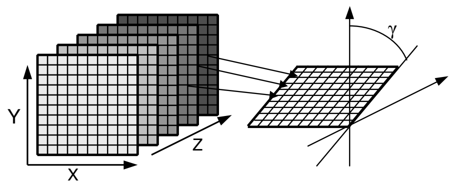



**Fig. 2.** Creating a projection

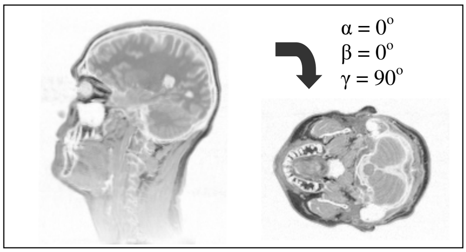



**Fig. 3.** A sample real projection for medical image

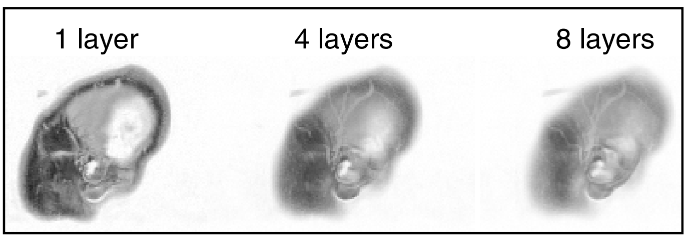



**Fig. 4.** A 3D view for an ear

also be asked for projection at any angle. To enable any rotation, client should give the parameters – angles:

– $\alpha(x, y)$,
– $\beta(x, z)$,
– $\gamma(y, z)$.

To create that projection for each pixel of this projection we should calculate its new coordinates (after rotation). The sample case is shown on Fig. 2 and Fig. 3 (sample images by: [8]).

To calculate new coordinates we use standard transformation for rotation:

$$\begin{array}{l} x' = x * \cos(\alpha) - y * \sin(\alpha) \\ y' = x * \sin(\alpha) + y * \cos(\alpha) \end{array} . \qquad (1)$$

This transformation we should process for three angles ($\alpha$, $\beta$, $\gamma$), given by client as parameters of http request. To make a 3D view the program needs to calculate a few of projections (5–10) and join them (Fig. 4).

### 3.1 Interpolation

The new coordinates are floating point values, but the block of data has indexes as integer values. To get a clear image we should interpolate the value of each pixel. One of the available methods to approximate the value of an intermediate point within the local axial rectangular prism linearly is trilinear interpolation (Fig. 5). The consultants endorsed the quality of this method is satisfactory.

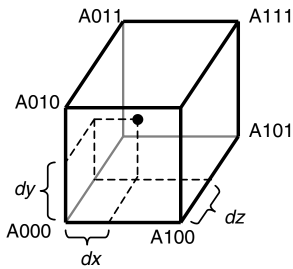



**Fig. 5.** New coordinates of a point after a rotation

### 3.2 Server-Side Creating and Processing an Image

At the beginning, the application loads selected CT results to a block of data and next can accept the http requests. The accepting requests consists of the following steps:

- handling a request by web server,
- creating a projection,
- creating an image file from projection (bmp or jpg format),
- posting image file to client.

Processing requests can be done simultaneously.

## 4 Performance Tests

To determine the response time and throughput for common Web technologies we made tests. There were applications written for Java servlets and ASP technology, running on the web servers (Apache 2.2.9 and IIS 6.0) under control of Linux 2.6 and Windows 2003 Server SP2. We measured the following parameters:

- time of projection processing (in the code),
- time of image file creating (in the code),

– minimal response time for a single request – using JMeter [7],
– maximal throughput of requests – using JMeter.

These tests were done for standard DICOM images (resolution 256x256 and 512x 512). Image file creating was tested for bitmap and JPEG 100% quality formats.

The test took place in a special environment. The Web server was installed on IBM HS20 Blade Server (2x Intel Xeon 2 core 2.8 GHz, RAM 4 GB, FSB 400 MHz) and client was installed on PC computer. These hosts were connected to 1 Gb Ethernet switch. Two operating systems – Linux and Windows Server – were installed on the server. PC client contained an instance of JMeter – testing application. The client PC and server were connected by a 1 Gb Ethernet switch.

## 5 The Results of Tests

The results of performance test are in Table 1. The data are also shown on Fig. 6, 7 and 8.

**Table 1.** Time performance of code – operating systems: Linux (L) and Windows (W)

| Format of image | Times of: [$ms$] | Java L | Java W | ASP W |
|---|---|---|---|---|
| 256 x 256 | projection | 30.5 | 81.2 | 42.1 |
| BMP | creation | 18.0 | 46.0 | 5.0 |
| | response time | 94.0 | 113.0 | 72.0 |
| | throughput [req/s] | 28.3 | 26.1 | 30.9 |
| JPG | creation | 18.0 | 46.0 | 8.0 |
| | response time | 85.0 | 105.0 | 57.0 |
| | throughput [req/s] | 32.5 | 19.5 | 42.5 |
| 512 x 512 | projection | 114.7 | 356.2 | 171.8 |
| BMP | creation | 63.0 | 124.0 | 15.0 |
| | response time | 370.0 | 450.0 | 360.0 |
| | throughput [req/s] | 7.5 | 6.6 | 4.6 |
| JPG | creation | 63.0 | 139.0 | 31.0 |
| | response time | 331.0 | 407.0 | 215.0 |
| | throughput [req/s] | 7.8 | 6.7 | 9.1 |

## 6 Summary

We tested two common web technologies (Java and ASP) running on Linux and Windows in terms of performance for medical image system. The fastest for our system is ASP/C# running on Windows. Although the projection calculating in this case isn't fastest, this technology performed the smallest request time and the highest throughput. The further work is to test such environments as PHP and Mono (.Net platform for Linux) and capability of hardware acceleration for imaging. The goal is to create a web-based transesophageal echocardiography simulator (TEE).

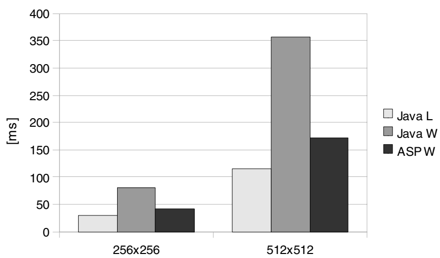



**Fig. 6.** Time performance of code for calculating a projection

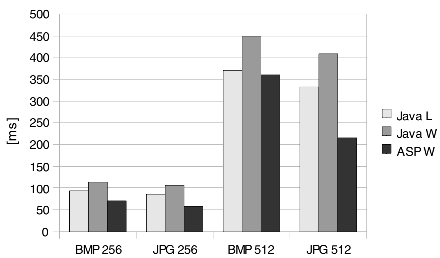



**Fig. 7.** Response times for single requests with a projection image

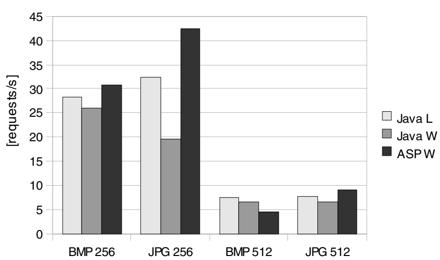



**Fig. 8.** The throughputs of DICOM image servers

## References

1. Sakusabe, T., Kimura, M., Onogi, Y.: On-demand server-side image processing for Web-based DICOM image display. In: SPIE. Proc. 2000, pp. 359–367 (2000)
2. Jianguo, Z., Jianyong, S., Yuanyuan, Y., Chenwen, L., Yihong, Y., Jin, J., Weihua, C., Kun, S., Guozhen, Z.: Image-Based Electronic Patient Records for Secured Collaborative Medical Applications. In: Engineering in Medicine and Biology Society, 2005, pp. 3218–3220 (2006)

3. Ancuta, P.N.: 3D Object Modeling and Visualization Software for Surgery Preoperative Plan. In: 6th Workshop on European Scientific and Industrial Collaboration on promoting Advanced Technologies in Manufacturing, WESIC 2008 (2008)
4. Rosset, A., Spadola, L., Ratib, O.: OsiriX: An Open-Source Software for Navigating in Multidimensional DICOM Images. Journal of Digital Imaging 17(3), 205–216 (2004)
5. Fernandez-Bayo, J., Barbero, O., Rubies, C., Sentis, M., Donoso, L.: Distributing medical images with internet technologies: a DICOM web server and a DICOM java viewer. Radiographics 20(6), 1817–1818 (2000)
6. Bui, A.A., Morioka, C., Dionisio, J.D., Johnson, D.B., Sinha, U., Ardekani, S., Taira, R.K., Aberle, D.R., El-Saden, S., Kangarloo, H.: OpenSourcePACS: an extensible infrastructure for medical image management. IEEE Transactions on Information Technology in Biomedicine 11, 94–109 (2007)
7. Apache JMeter, http://jakarta.apache.org/jmeter/
8. Sample images, http://rsb.info.nih.gov/ij/download/sample-images.zip

# Efficiency Analysis of the Server-Side Numerical Computations*

Adam Piorkowski and Daniel Plodzien

Department of Geoinfomatics and Applied Computer Science,
AGH University of Science and Technology, Cracow, Poland
pioro@agh.edu.pl
http://www.agh.edu.pl

**Abstract.** The Web client architecture becomes very popular distributed software solution. In this paper the leading Web technologies (CGI, PHP, Java, .Net) are described and compared in terms of efficiency of numerical computations. Such considerations are relevant to issues related to e.g. the processing of images or geoinformatics. The synthetic tests covered a matrix multiplication with different sizes for integers and floating and the use of common mathematical functions – power, square root, sine and tangent. The processing times and throughputs are measured experimentally.

## 1 Introduction

The Web client architecture becomes very popular distributed software solution. Centralized control, easy software maintenance and isolation of data – these are the main advantages of this model. There is no need to install a special software – a simple Web browser can be a client for these systems. With the increase in computing power of servers, there are new opportunities for applications, e.g. simulations, calculations, geocomputing or image processing. Parallel computing is a method to improve efficiency for time-consuming calculations. Nowadays there are many CPUs in internet servers, the load balancing for these servers is available, so in some cases clusters aren't necessary. The efficiency of the calculations depends not only on equipment but also on the software environment. The main efficiency factors for internet servers are the request time and the throughput. The goal of this article is to experimentally evaluate these factors for numerical calculations in the leading Web technologies (CGI, PHP, Java servlets and Asp/C#).

There have been numerous studies evaluating the Web technologies performance. In 2008, Scott Trent et al. [1] tested and compared the performance of PHP with FastCGI extension and JSP technologies for various common algorithms (Fibonacci, Levenshtein, Quick Sort). The size of problem was not big

* This work was financed by the AGH – University of Science and Technology, Faculty of Geology, Geophysics and Environmental Protection as a part of statutory project number 11.11.140.561.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 225–232, 2009.

and algorithm runtime was the smaller part of the request. The tests were done for Apache and Lighttpd http servers.

Cecchet et al. presented benchmarks for two applications: online bookstore and auction site [2]. These application were connected to databases and stored large data (pictures) as files in file system. The performance was measured for three technologies: PHP, Java Servlets and Enterprise Java Beans. The throughput and CPU utilization were selected as factors.

Titchkosky et al. experimentally evaluated the impact of dynamic content technologies (Perl, PHP and Java servlets) on Web server performance [3]. The tests were done for static and dynamic content of pages with or without database access.

Apte et al. presented another study [4] on performance comparison of dynamic Web platforms. CGI, FastCGI, JSP and Java servlets were under a scope. Authors proved that FastCGI was the best performing of the technologies compared.

An efficiency analysis for numerical computations in component environments was presented by Kowal et al.[5]. In this study a performance for distributed processing of seismic wave field modeling was compared for hardware independent platforms that use managed code technique: Java VM, MS .Net and Mono. This is an example for time-consuming calculations.

## 2 Dynamic Web Platforms

From variety of technologies supporting server-side scripting we chose following solutions: CGI, PHP, Java Servlets, ASP.NET and Mono. Most of our experiments involve Apache Web Server with a collection of appropriate modules which extends the main functionality of Web server. For our purposes we used three modules: mod_cgi which provides Apache server with capability of execution the CGI scripts, mod_php which provides Apache server with functionality of processing PHP scripts and mod_mono module, which allows for hosting ASP.NET scripts by use of the MONO runtime environment. Another module is Apache Tomcat, which implements Java Servlet and JavaServer Pages specifications from Sun Microsystems and allows for hosting Servlet components and JSP scripts by use of Java Virtual Machine runtime environment.

CGI (Common Gateway Interface) is the earliest technology which enables dynamic request processing and non-static content creation. CGI scripts can be written in any programming language.

PHP (Hypertext Preprocessor) is a scripting language designed for producing dynamic Web pages which has become one of the most popular server-side scripting language for last few years.

IIS (Internet Information Services) is a set of Internet-based services for use with Microsoft Windows. One of the services is a Web server, which make it possible to execute ASP.Net scripts by use of .Net CLR (Common Language Runtime) runtime environment.

# 3 Testing Environment and Methodology

The test took place in a special environment. The Web server was installed on IBM HS20 Blade Server and client was installed on PC computer. These hosts were connected to 1 Gb Ethernet switch. There were installed two operating systems on the server – Linux and Windows Server. PC client contained an instance of JMeter – testing application.

## 3.1 Software Environment

The server worked under control operating systems:

- Linux 2.6 Fedora 8,
- Windows 2003 Server SP2.

The following compilers / environments were used:

- gcc Linux: 4.1.2,
- gcc Win32: 3.4.2 (mingw special),
- Visual C++ 2008 EE,
- Sun Java Linux/Win32: 1.6.0.11,
- MS .Net 2.0,
- Mono 2.2.

All C compilers were switched to -O2 option (optimize for speed). We use the following Web software:

- Apache 2.2.9,
- IIS 6.0,
- Tomcat 6.0.18,
- PHP 5.2.6.

## 3.2 Hardware Environment

Our tests were performed on IBM HS20 Blade Server:

- CPU: 2 x Dual-core Intel Xeon Processor, 2.8 GHz,
- CPU L2 Cache (shared): 4 MB,
- RAM: 32 GB, FSB 400 MHz.

As a client computer we used a PC computer, with Dual-core Intel Xeon Processor, 2.8 GHz, 1 GB RAM. This computer and Blade server were connected by a 1 Gb Ethernet switch.

## 3.3 Synthetic Tests for Math Operations

To indicate numerical calculations performance we wrote synthetic math applications.

**Multiplication of Matrixes.** During multiplication of matrixes CPU makes a big number of integer or float operations. These applications allow to mark speed of potential common numeric calculations. The first test was to multiply two matrixes 10x10 dimension, 100,000 times. The second – bigger matrixes (100x100), but only 100 times.

**Simple Math Functions.** We tested performance for four common math functions: power, square root, sine and tangent in numerous iterations. To avoid a compiler-specific optimization, the total sum from each operation is calculated. The main parts of sample code for C language are presented below.

```
// Code for power/square root/sine/tangent function tests
for (i=0; i<n; i++)
    powSum += pow(i%10, i%10);              // power
    sqrtSum += sqrt(i);                     // square root
    sinSum += sin((i%360)*M_PI/180.0);      // sine
    tanSum += tan((i%90)*M_PI/180.0);       // tangent
```

### 3.4 Implementation and Test Methodology

We measured three parameters for each case:

- time performance of code,
- response time for single request (minimal),
- throughput (requests/sec).

The first parameter is the time performance of code in program, indicated by system functions. To get second and third parameters we used a standard program – JMeter [6].

### 3.5 Test Categories

The following categories of tests took a place:

- M10 I – multiplication of matrixes 10x10, integer, 100,000 iterations,
- M10 D – multiplication of matrixes 10x10, double, 100,000 iterations,
- M100 I – multiplication of matrixes 100x100, integer, 100 iterations,
- M100 D – multiplication of matrixes 100x100, double, 100 iterations,
- Pow – a sum of powers for 1,000,000 iterations,
- Sqrt – a sum of square roots for 10,000,000 iterations,
- Sin – a sum of sine values for 1,000,000 iterations,
- Tan – a sum of tangent values for 1,000,000 iterations.

The tests were done under control of Linux (marked 'L') and Windows (marked 'W') operating systems for PHP, Java and CGI (gcc). The code for CGI with VC++ 2008 was runable only under Windows. The .Net applications used .Net framework under Windows and Mono under Linux.

## 4 Experimental Results

The tests were designed so that the network connection overhead is very small compared to the time of calculation. The results proved that minimal response time is very close to time performance of code. Therefore we present the second one only. The Table 1 contains results of tests for matrix multiplying (size 10x10) and math functions, the Table 2 – throughputs for these cases. The experimental results of response times for matrix multiplications are shown on the Fig. 1. The next figure (Fig. 2) shows times for math functions. Figure 3 reports throughput for processing the matrix multiplication. The last one (Fig. 4) shows throughput for iterations that call math functions.

**Table 1.** Time performance of code [ms] – Linux (L) and Windows (W)

| Test | CGI-GCC | CGI-VC | PHP | Java | ASP/C# |
|---|---|---|---|---|---|
| L M10 I | 260 | — | 52500 | 647 | 2047 |
| L M10 D | 290 | — | 56600 | 528 | 2111 |
| L M100 I | 290 | — | 53900 | 440 | 1584 |
| L M100 D | 250 | — | 57900 | 438 | 1421 |
| L Pow | 120 | — | 926 | 523 | 422 |
| L Sqrt | 160 | — | 5660 | 151 | 416 |
| L Sin | 90 | — | 1033 | 229 | 107 |
| L Tan | 120 | — | 1054 | 260 | 138 |
| W M10 I | 281 | 140 | 56200 | 857 | 1151 |
| W M10 D | 281 | 156 | 56200 | 844 | 1265 |
| W M100 I | 296 | 250 | 54300 | 797 | 921 |
| W M100 D | 250 | 218 | 54600 | 671 | 937 |
| W Pow | 187 | 17 | 921 | 578 | 234 |
| W Sqrt | 156 | 250 | 5276 | 171 | 343 |
| W Sin | 328 | 75 | 1004 | 156 | 93 |
| W Tan | 359 | 96 | 1043 | 230 | 140 |

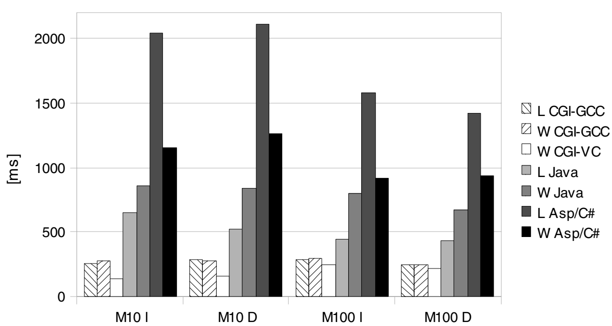



**Fig. 1.** Time performance of code for matrix multiplications

**Table 2.** Throughput [requests/s] – Linux (L) and Windows (W)

| Test | CGI-gcc | CGI-VC | PHP | Java | ASP/C# |
|---|---|---|---|---|---|
| L M10 Int | 7.7 | — | 3.5 | 2.9 | 1.1 |
| L M10 D | 8.2 | — | 3.4 | 3.4 | 0.9 |
| L M100 I | 6.5 | — | 3.4 | 3.9 | 1.5 |
| L M100 D | 7.5 | — | 3.3 | 5.0 | 1.5 |
| L Pow | 19.2 | — | 1.4 | 6.0 | 6.7 |
| L Sqrt | 11.4 | — | 0.1 | 11.6 | 6.0 |
| L Sin | 25.6 | — | 1.3 | 13.9 | 28.7 |
| L Tan | 21.3 | — | 1.2 | 12.5 | 22.3 |
| W M10 I | 7.6 | 13.0 | 3.7 | 2.5 | 2.0 |
| W M10 D | 8.4 | 12.4 | 3.6 | 2.6 | 1.8 |
| W M100 I | 6.3 | 7.5 | 3.4 | 2.9 | 2.3 |
| W M100 D | 7.5 | 10.2 | 3.4 | 3.0 | 2.1 |
| W Pow | 14.6 | 80.2 | 2.4 | 5.3 | 13.2 |
| W Sqrt | 11.7 | 8.8 | 0.3 | 13.1 | 6.6 |
| W Sin | 9.5 | 21.0 | 2.3 | 16.1 | 29.6 |
| W Tan | 8.9 | 17.9 | 2.1 | 10.1 | 22.3 |

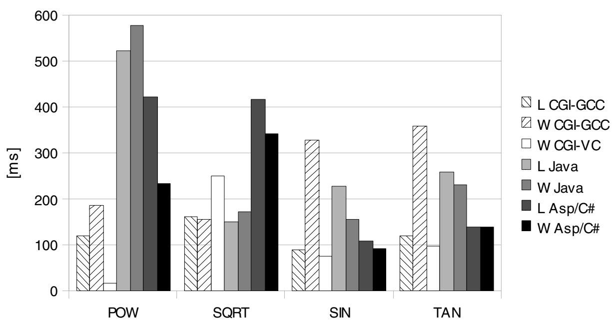



**Fig. 2.** Time performance of code for math functions

## 5 Summary

The best results reached CGI – the shortest times of processing and the maximal throughputs. In most cases the winner was the code compiled and linked by Visual C++. The results for PHP technology are clearly worse than others, so we decided to omit them on the plots. It's hard to choose, which component platform (Java or ASP/C#) is faster – the results for matrix multiplications are better for Java, but performance of simple math functions is better for C#. There is simple correlation between processing times and throughput – we expected that the throughput for component environments should be better than CGI (the one

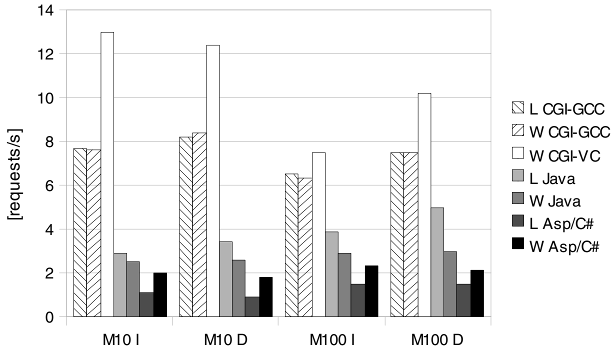



**Fig. 3.** Throughput for matrix multiplications

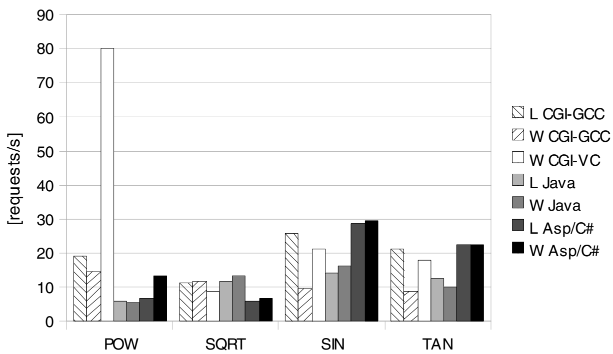



**Fig. 4.** Throughput for math functions

of reasons for using them instead of CGI), but CPU resource is a bottleneck for numerical computations.

## 6 Conclusions

We compared four leading Web technologies (CGI, PHP, Java and ASP) in terms of efficiency of numerical computations. These computations are common in such numerical domains as geocomputing or image processing. The most efficient technology in this case is CGI/FastCGI, especially with Visual C++ compiler. The PHP performance is the worst in context of numerical computing. In component platforms (Java, ASP/C#) the choice between Java and Asp/C# isn't easy, but these technologies give environments for fast code developing and are very popular now. The future work should focus on performance of special math libraries for component technologies.

## References

1. Trent, S., Tatsubori, M., Suzumura, T., Tozawa, A., Onodera, T.: Performance Comparison of PHP and JSP as Server-Side Scripting Languages. In: Issarny, V., Schantz, R. (eds.) Middleware 2008. LNCS, vol. 5346, pp. 164–182. Springer, Heidelberg (2008)
2. Cecchet, E., Chanda, A., Elnikety, S., Marguerite, J., Zwaenepoel, W.: Performance Comparison of Middleware Architectures for Generating Dynamic Web Content. In: Endler, M., Schmidt, D.C. (eds.) Middleware 2003. LNCS, vol. 2672. Springer, Heidelberg (2003)
3. Titchkosky, L., Arlitt, M., Williamson, C.: A performance comparison of dynamic Web technologies. ACM SIGMETRICS Performance Evaluation Review archive 31(3), 2–11 (2003)
4. Aptea, V., Hansen, T., Reeserb, P.: Performance comparison of dynamic Web platforms. Computer Communications 26(8), 888–898 (2003)
5. Kowal, A., Piorkowski, A., Danek, T., Pieta, A.: Analysis of selected component technologies efficiency for parallel and distributed seismic wave field modeling. In: International Joint Conferences on Computer, Information, and Systems Sciences, and Engineering (CIS2E 2008), December 5-13 (2008)
6. Apache JMeter, http://jakarta.apache.org/jmeter/

# Adaptive Approach to Network Security

Bogdan Ksiezopolski[1], Zbigniew Kotulski[2,3], and Pawel Szalachowski[1]

[1] Institute of Computer Science, Maria Curie-Sklodowska University, pl. M. Curie-Sklodowskiej 5, 20-031 Lublin, Poland
[2] Institute of Fundamental Technological Research of PAS, Swietokrzyska 21, 00-049 Warsaw, Poland
[3] Institute of Telecommunications of WUT, Nowowiejska 15/19, 00-665 Warsaw, Poland

**Abstract.** The security of information exchange between the parts in the teleinformatic infrastructure is one of the crucial topics. During the protecting the infrastructure of the organization, one can use a lot of mechanisms which are often based on the cryptographic primitives. Traditionally, the security officers model the protection system to be as strong as it is possible. However, the level of protection of information is often much higher than it is necessary to meet potential threats. Since the level of security strongly affects the performance of the whole system, the excessive protection decreases its reliability and availability and, as a result, its global security. The appropriate security level can be estimated by means of different quality of protection models. In this paper we are going to present the approach how to introduce the adaptability to the network. We are basing on the adaptable security model for dynamic environment which calculates the protection level by means of the risk management processes. The analysis is assisted by the security management tool (SPOT) which visualizes and optimizes the adaptable model mentioned above. Finally, we present the case study introducing the network adaptability of the cryptographic protocol.

## 1 Introduction

Nowadays, advanced teleinformatic technologies provide us with a wide range of possibilities of development of industry institutions and public services. Emphasis is put on the development of well-available, mobile information services called e-anything, like e-government, e-money, and e-banking. Implementation of these services would be connected with the choice of proper level of security of the information sent between parties of protocols [1]. One of the important problems is establishing an appropriate level of information security, represented by security services in a given protocol. Each use of any Internet services is connected with information exchange, which in the case of successful attack, causes different threats to the whole process. This problem can be solved by estimating the security level for each phase of the protocol [2]. Such an approach is only a partial solution, because during the particular phase of the protocol, one can send information on different levels of threats. Traditionally, the aim has been to

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 233–241, 2009.

provide the strongest possible security. However, the use of strong mechanisms may deteriorate the performance of a device with limited resources and pave the way for new threats such as resource exhaustion. Finally, it decreases system efficiency, availability and introduces redundancy. Another effect of overestimation of security mechanisms is increasing the system complexity, which later influences implementation of a given project, imposing restrictions that decrease their functionality. The adequate solution in such a case is the introduction of adaptable (or scalable) security model for the protocols, which can change the security level depending on particular conditions that take place at a certain moment, and in given external conditions.

In this paper, we are use Quality of Protection (QoP) models [3,4,5,6,7]. In the literature one can find only a couple of articles about QoP because this security topic is one of the newest approaches. Lindskog tries to extend security layer in a few Quality of Service (QoS) architectures [3]. Unfortunately, the described methods are limited to the confidentiality of the data. These methods are based on different configurations of the cryptographic modules. Ong in [5] presents QoP mechanisms, which define security levels depending on security parameters. These parameters are: a key length, the length and contents of an encrypted block of data. Schneck i Schwan [4] proposed the adaptable protocol concentrating on the authorization. By means of this protocol one can change the version of the authorization protocol which finally changes the parameters of the asymmetric and symmetric ciphers. The Sun [6] creates QoP models based on the vulnerabilities analysis which are represented by the attack trees. The leafs of the trees are described by means of the special metrics of security. These metrics are used for describing individual characteristics of the attack. Unfortunately, the majority of the QoP models can be realized only for the three main security services: confidentiality, integrity and authorization. In the article [7] Ksiezopolski introduces mechanism for adaptable security which can be realized for all the security services. In Section 2 we briefly present the model, which Ksiezopolski introduces in [7].

In this article we present the process of introducing the adaptability to the network security. We are basing on the adaptable model [7] thanks to which one can calculate the different version of the same protocol which can be realized on different level of security. The configuration of the model was prepared by means of the SPOT application [8] which is the visualization of the model. In the paper we have defined three versions of the TLS cryptographic protocol which realized security on different level. Finally, the theoretically calculated versions of TLS protocol have been implemented and tested in the laboratory.

## 2 Model of Adaptable Security

The realization of an electronic process strongly depends on the proper level of security. While designing such a process, the security mechanisms are usually overestimated according to real risk.

The security level of an electronic process depends on several factors. This level can be modified by the choice of security elements applied in a protection system. In the model of the adaptable security [7], one can suggest an analytical expression to calculate the security level; its numerical value is a function of three primary parameters:

- Protection level of security service ($L$);
- Probability of incident occurrence ($P$);
- Impact of a successful attack ($\omega$).

Each of the parameters is calculated for all cryptographic protocols, all subprotocols within these protocols, and all steps within these subprotocols.

The first parameter defines the protection level for a given cryptographic service in a given step of a subprotocol. It is the sum of the effects of the chosen security elements which guarantee security of a given service.

The second parameter represents a probability of incident occurrence on the security service. This parameter is associated with the risk of electronic process [9]. The choice of the configuration of security elements has the biggest influence on this parameters. The security elements, which we can use to realize security requirements are represented by trees. The selection of leaves refers to the selection of specific security elements which will be used in the protocol. The calculation of this protocol is complex and can not be presented in this article but the details can be found in another article [10].

The impact of a successful attack is the third parameter which influences the security level of the process. We calculate it, as previously, for each service in each step. For calculation we use direct and indirect parameters,which are presented below:

- **The direct parameters:**

  $LZ$ – assets gained during a successful attack on given security elements (100% is the compromise of the whole protocol);

  $F$ – financial losses during a successful attack on given security elements (100% is the total financial loss).

- **The indirect parameters:**

  $\alpha$ – necessary financial costs for repairing the damages gained during a successful attack (100% is the maximal cost);

  $\beta$ – losses of the value of the company shares or the company reputation (100% is the maximal market loss).

Finally the impact of the successful attack is calculated by the formula presented below:

$$\omega = \frac{LZ}{3}(F + \beta + \alpha) \ . \tag{1}$$

### 2.1 Security Level ($F_S$)

The global security level expresses the security of the whole cryptographic protocol. We calculate this factor according to the formula:

$$F_{\mathrm{S}} = \frac{1}{a}\sum_{i=1}^{a}\frac{1}{b_i}\sum_{j=1}^{b_i}\frac{1}{c_{ij}}\sum_{x=1}^{c_{ij}}\left(L_{ij}^{x}\right)^{Z}\left[\left(1-\omega_{ij}^{x}\right)\left(1-P_{ij,\mathrm{ALL}}^{x}\right)\right] \tag{2}$$

where:

$F_S$ is the security level realized by a given version of cryptographic protocol, $F_S \in (0,1)$;
$i$ is the number of subprotocols in a given protocol;
$j$ is the number of steps in a given subprotocol;
$x$ is the number of specific security services;
$\omega_{ij}^{x}$ is the weight describing an average cost of loses after a successful attack on a given service, $\omega \in (0,1)$;
$L_{ij}^{x}$ is the value of a protection level for a given service, $L \in (0,1)$;
$P_{ij}^{x}$ is the probability of an attack on a given service, $P \in (0,1)$;
$Z$ is the scalability parameter for security elements, $Z \in (0,10)$.

### 2.2 SPOT: Security Protocol Optimization Tool

Security Protocol Optimization Tool (SPOT) [8] is the application which main function is visualization of the adaptable model [7]. By means of this tool we can create versions of the given protocol, compare these versions and visualize the results. It is also designed to be portable, user-friendly and compatible with other elements. The important feature of the SPOT is optimization module, which is capable of generating all states of the given protocol and show the optimal states according to the user's preferences. Unfortunately, the description of the SPOT architecture and the modules takes too much space so we can not include it in this article but these information are available in another article [8].

## 3 The Version of the TLS Protocol – Adaptable Model

In this article we would like to focus on the adaptive approach to network security. In this section we would like to present the study of applying adaptive security for one of the most often used cryptographic protocols – TLS [11]. The organizations can use this protocol in many situations, it can be tunneling of VPN transmission, transferring data backup to the data warehouse or just HTTPS connection. The organization staff exchange the information by means of different devices and very often these devices are mobile. In the case of the devices with limited resources (mobile phones, PDA, sensors) speed and efficiency is crucial for their stable work. The security methods used during transporting the data between the parts are crucial in terms of efficiency. One can notice that information transmitted by means of the TLS protocol have different character. Some of them are crucial for the organization (transaction information, plans of investment) but some of them are non-sensitive data which are exchanged during every workday.

The security mechanisms in the TLS protocol can be configurated in different ways. During the transmission one can ensure confidentiality of the data by means of different symmetric ciphers, it could be [11]: 3des-cbc, blowfish-cbc, aes128-cbc, aes192-cbc, aes256-cbc, aes128-ctr, aes192-ctr, aes256-ctr,cast128-cbc, arcfour, arcfour128, arcfour256, cast128-cbc. The connections integrity is realized by means of different hmac functions and it can be [11]: hmac-md5, hmac-sha1, umac-64, hmac-ripemd160. The transmitted data can be protected by the TLS protocol with different combinations of symetric cipher and hmac functions.

In the article we use adaptable model [7] to calculate different versions of the TLS protocol. We use SPOT applications thanks to which we can easily prepare different versions of the protocol. Unfortunately, the detailed description of the whole process of configuring and using the adaptable model takes more than 5 pages so we would not presented it in the article.

### 3.1 Version 1

The first TLS protocol version is prepared for the scenario when the most crucial data is exchanged and required security services are: confidentiality and integrity. The detailed values of adaptable model parameters for all versions of the TLS protocol are presented in the Table 1.

**Table 1.** The adaptable model parameters for the TLS protocol

| | $F$ | $\alpha$ | $\beta$ | $P$ | $\omega$ | $F_S$ | Cipher |
|---|---|---|---|---|---|---|---|
| *Version 1* | | | | | | | |
| **Confidentiality** | 0.96 | 0.92 | 0.96 | 0.40 | 0.85 | 0.15 | 3DES |
| **Integrity** | 0.97 | 0.91 | 0.92 | 0.41 | 0.65 | 0.15 | HMAC-RIPEMD160 |
| *Version 2* | | | | | | | |
| **Confidentiality** | 0.72 | 0.70 | 0.75 | 0.40 | 0.65 | 0.25 | 3DES |
| **Integrity** | 0.67 | 0.74 | 0.72 | 0.41 | 0.50 | 0.25 | HMAC-RIPEMD160 |
| *Version 3* | | | | | | | |
| **Confidentiality** | 0.72 | 0.70 | 0.75 | 0.66 | 0.65 | 0.19 | RC4 |
| **Integrity** | 0.67 | 0.74 | 0.72 | 0.50 | 0.50 | 0.19 | HMAC-MD5 |

One can notice that in the first protocol version the global financial losses during a successful attack ($F$) and necessary financial costs for repairing the damages gained during a successful attack ($\alpha$) and losses of the value of the company reputation ($\beta$) are almost maximum (1 is maximum value). As the result of these characteristics of the transmitted data the impact of successful attack ($\omega$) is on the high level. These parameters refers to the confidentiality and integrity of information exchanged in TLS and are crucial for the organization. In the adaptable model we have to choose specific ciphers for ensuring confidentiality and integrity. These parameters influence the probability of incident occurrence ($P$). For the first version we choose symmetric cipher *3DES* and hmac function *HMAC-RIPEMD160*. Finally we calculate the global security level ($F_S$) and for the first version it equals 0.15 (maximum is 1).

### 3.2 Version 2

The second version of the protocol assumes that transmitted data would have non-sensitive character for the organization. The firm doesn't want to resign from the security requirements defined in the first version and they still want to guarantee the confidentiality and integrity of the transmitted data. In the adaptable model we change the parameters which refer to the type of data ($F$, $\alpha$, $\beta$). As the result of changing these parameters the impact of successful attack ($\omega$) is on the medium level. In the second protocol version we don't change the symmetric ciphers and hmac functions and as the result of that the level of probability of incident occurrence is the same as in the first version. In the last step we calculate global security level and because the impact of successful attack is on medium level the value of global security is higher than in the first version and equals 0.25. The first version of the TLS protocol indicates the minimum level of global security which guarantees the required security for the processes. The second version of the protocol has higher level of global security than it is needed so we can state that the security mechanisms are overestimated according to the type of the transmitted data. According to the adaptable model we can change the details about the security mechanism but finally calculated global security level ($F_S$) must be higher than in the first version of the protocol.

### 3.3 Version 3

In the third version of the protocol we change the details about the security mechanisms which ensure the confidentiality and integrity of transmitted data. The parameters which define the impact of successful attack don't change because the type of information is the same. In the adaptable model we find the configuration of security mechanisms but in case of transmitting non-sensitive type of data the probability of incident occurrence can be higher. The configurations calculated by the model show that the probability of incident occurrence will be higher when we choose as the symmetric cipher *RC4* and as the hmac function *HMAC-MD5*. This version of the protocol has adequate level of global security ($F_S$) because it equals 0.19 and is still higher than in the first version of the protocol.

Comparing these three versions of the TLS protocol we can state that it is possible to change the configuration of the protocol with ensuring adequate for the actual risk global security. The version of the protocol which is adequate to specific requirements for the processes can be calculated by means of the adaptable model. By changing the version of TLS protocol one can increase the efficiency of the process. In the following section we would like to present the case study of the realization of the described versions of TLS protocol. Thanks to these results we are able to estimate the efficiency of the systems which takes part in the transmission.

## 4 Case Study

In this section we would like to present the results of transferring the data by means of the versions of TLS protocol described in the section above. We simulate scenario when the organization stores the data in the data warehouse. During the test the organization stored the 1 GB packet of data. The communication channel is protected by means of TLS protocol. The most important parameters which influence the efficiency of the system taking part in transmission are the time of transmission and the CPU load and memory usage.

The tests was prepared by two computers class PC connected by 100 MB UTP cable. The processors in the computers are Intel Celeron 2.00 GHz with 128 KB cache, the bogomips of the CPU is 3986.44. The memory of the PC is 1 GB. The system of the PCs is Linux Debian 4.0 with specially prepared configuration which guarantees that during the tests no extra services and applications, except the testing one, will run in the system.

The changing of the TLS protocol configuration can be realized by means of two possibilities. It could be realized in off-line mode and it means that the configurations are changing before the transmission of the data. Another possibility is on-line mode and then the switching process is realized during transmission. The results presented in the article were prepared in the off-line mode.

In the Table 2 we present the test results for three versions of the TLS protocol described in the section above. The first and the second version have the same results because the differences between these versions refer only to the parameters influencing the impact of a successful attack. In these versions the sets of used ciphers are the same.

The crucial results are noticed while comparing the second and the third version of the TLS protocol. In the third version of the protocol we increase the probability of incident occurrence for this process but we acquire the efficiency of the system. The time of transmission of the same 1 GB data takes 436 seconds in the second version of protocol, while in the third version of the same protocol

**Table 2.** The test results for three version of the TLS protocol

| Ciphers | Time [s] | CPU load [%] | RAM [KB] |
|---|---|---|---|
| *Version 1* | | | |
| **3DES + HMAC-RIPEMD160** | 436 | 42 | 13412 |
| *Version 2* | | | |
| **3DES + HMAC-RIPEMD160** | 436 | 42 | 13412 |
| *Version 3* | | | |
| **RC4 + HMAC-MD5** | 269 | 33 | 13404 |

takes only 269 seconds. In the third version we decrease the time transmission to 38%. The second crucial system parameter is CPU load. In the third version the system needed 33% of CPU load when the second version needed 42%. This is another parameter which indicates that the third version is more efficient than the second version. The CPU load is especially important when the transmission is realized by means of devices with limited resources (mobile devices) because the battery life directly depends on the CPU load. The memory usage in the compared versions of the protocol is almost on the same level.

## 5 Conclusion

In the article we present the adaptable approach to network security. The organizations in the industry and public services exchange a huge number of data every day. The usage of the higher than is require protection level during data transmission leads to the decreasing efficiency of the devices from which the transmissions is realized.

In the article we are shown the usage of adaptable model [7] thanks to which one can calculate the different security versions of the same cryptographic protocol. We are present different versions of the TLS protocol which was obtained in the adaptable model. It is possible to create two versions of the protocol for the different kind of data and be sure that they are protected adequately to the actual potential risk. The theoretical model results were verified by the test results carried out in the laboratory. Thanks to usage the appropriate to the risk version of the TLS protocol one can decrease the transmission time of 1 GB data to 38% and decrease CPU load from 42% to 33%. These results are especially important when the transmission is realized by the mobile devices because the CPU load directly influences the battery life.

## References


1. Merabti, M., Shi, Q., Oppliger, R.: Advanced security techniques for network protection. Computer Communications 23, 151–158 (2000)
2. Lambrinoudakis, C., Gritzalis, S., Dridi, F., Pernul, G.: Security requirements for e-government services: a methodological approach for developing a common pki-based security policy. Computer Communication 26, 1873–1883 (2003)
3. Lindskog, S.: Modeling and Tuning Security from a Quality of Service Perspective. PhD dissertation, Department of Computer Science and Engineering, Chalmers University of Technology, Gteborg, Sweden (2005)
4. Schneck, P., Schwan, K.: Authenticast: An Adaptive Protocol for High-Performance. Secure Network Applications, Technical Report GIT-CC-97-22 (1997)
5. Ong, C., Nahrstedt, K., Yuan, W.: Quality of protection for mobile applications. In: IEEE International Conference on Multimedia & Expo. (2003)
6. Sun, Y., Kumar, A.: Quality od protection(qop): A quantitative methodology to grade security services. In: 28th confrence on Distributed Computing Systems Workshop, pp. 394–399 (2008)

7. Ksiezopolski, B., Kotulski, Z.: Adaptable security mechanism for the dynamic environments. Computers & Security 26(3), 246–255 (2007)
8. Szalachowski, P., Ksiezopolski, B., Kotulski, Z.: SPOT: Optimization tool for the adaptable security mechanism for dynamic environments (submitted to the publication, 2009)
9. Gerber, M., Solms, R.: Management of risk in the information age. Computers & Security 14, 16–30 (2005)
10. Ksiezopolski, B., Kotulski, Z.: On a probability modeling of incidence occurrence in electronic processes. In: 7th NATO Regional Conference nn military communications and information systems, pp. 297–305 (2005)
11. Dierks, T., Rescorla, E.: The Transport Layer Security (TLS) Protocol Version 1.2, RFC 5246, IETF (2008)

# PROFINET I/O Network Analyzer

Rafał Cupek[1], Markus Bregulla[2], and Łukasz Huczała[3]

[1] Silesian University of Technology, Institute of Informatics
rcupek@polsl.pl
[2] Fachhochschule Ingolstadt
bregulla@fh-ingolstadt.de
[3] Silesian University of Technology
lukasz.huczala@gmail.com

**Abstract.** This chapter describes PROFINET I/O network analyzer that was created as an open project realized in Institute of Informatics, Silesian University of Technology within cooperation with University of Applied Sciences Ingolstadt. Authors provide description of a PROFINET I/O analyzer idea which was based on joined analysis operation modes: active industrial network structure scanning and passive listening to a real-time communication. The base information about industrial PROFINET I/O structure, communication protocol and the core structure of realized network analyzer are presented also.

## 1 Introduction

The real-time industrial network area is now changing from solutions based on standard serial communication into area of dedicated real-time Ethernet based solutions. One of examples is PROFINET I/O industrial network which is the successor of very popular PROFIBUS low level industrial network. Another fact is the increasing size of new applications in the area of industrial networks. Distributed industrial control systems have become more and more complicated. New functional requirements for horizontal and vertical communication are given. Despite these facts there are very few solutions which can help engineers to startup and maintain industrial networks based on real-time Ethernet. There are some general purpose network analyzers, but very few solutions are dedicated to low level Ethernet based industrial networks. That fact was the impulse to start the PROFINET I/O analyzer project. This project was realized as the cooperation of two technical universities Silesian University of Technology and University of Applied Sciences Ingolstadt with PNO Poland (Profibus Nutzer Organization) support. On the basis of described project another research in horizontal communication in distributed real-time control systems has been started. This project has started the ability for next research works in range of vertical non-invasion data acquisition solutions used for vertical communication in industrial computer systems also. The project idea and PROFINET I/O communication principles are described in this article.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 242–251, 2009.

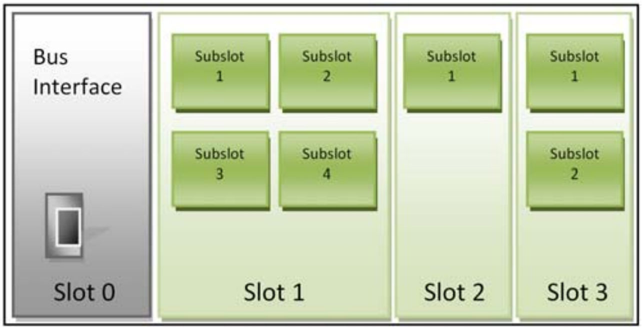



**Fig. 1.** PROFINET I/O device model

## 2 PROFINET I/O Low Level Communication Network

PROFINET I/O is an Ethernet based industrial standard developed by PROFIBUS International for creating integrated and compact automation systems [7,8]. The PROFINET technology is applicable in two areas. The first, PROFINET I/O, covers the domain of field devices, the second solution is called PROFINET CBA (Component Based Automation) and it shows its strengths in communication between controllers (PLCs). As this article covers the topic of PROFINET I/O network analyzer the base aspect of PROFINET I/O communication will be presented.

In distributed real-time systems the data from physical inputs and outputs of field devices is cyclically read by the PLC. PROFINET I/O standard defines an I/O-Device, patterned on PROFIBUS DP model, which consists of a slot and a channel [4]. The model allows for configuration of both modular and compact filed devices. As compared to PROFIBUS, the submodule layer has been added in order to accommodate the flexibility of modern field devices (Fig. 1). Modules are addressed via slots while submodules are addressed via subslots. Slots/subslots are equipped with input and output channels to exchange process data.

Every I/O-Device has a unique id which is a 32-bit number divided into a 16-bit producer id and a 16-bit device id part. The producer id is granted by Profibus International when device id is assigned by the producer according to its design conditions.

Device's parameters are described in GSD file (General Station Description), which is based on XML standard [5]. The file provides all needed information:

- I/O-Device properties (e.g. communication parameters);
- Installed modules (a type and quantity);
- Configuration data for every module;
- Modules' parameters;
- Error messages' text for diagnosis purposes.

GSD files are distributed together with devices. After uploading the device data into configuration tool, user must configure every device by selecting proper


**Fig. 2.** PROFINET I/O communication channels

modules and setting connection parameters. The created configuration data, in form of a project, is then being downloaded to the controller. The controller, by means of acyclic parameterization services, sends the configuration to every device included in the project.

PROFINET standard distinguishes three types of devices called "PROFINET roles":

- I/O-Controller: a PLC, at which a system control program is executed;
- I/O-Device: a filed device assigned (remotely) to the I/O-Controller;
- I/O-Supervisor: programming device (or PC station) with diagnostic functions.

Data can be transferred between PLC (I/O-Controller) and field devices (IO-Devices) through the following channels (Fig. 2):

- Cyclic I/O data via real-time channel;
- Alarms via real-time channel;
- Parameterization, configuration and diagnostic data via standard channel based on UDP/IP.

PROFINET IO standard uses different network layers in communication process, which differ in efficiency:

- Non-time-critical data, such as parameters, configuration data, connections information, is sent by means of TCP/UDP and IP channel.
- For transferring time-critical data PROFINET uses a real-time channel – SRT (Synchronous Real Time). The channel is implemented in form of software in PLCs.
- Applications requiring time-driven communication are provided with "Isochronous Real Time" channel (IRT), which secures an impulse precision at the level of 1 micro second with clock rate of 1 ms.

The standard TCP/IP protocols [3] are used for non-real-time services (Fig. 3). UDP/IP (User Datagram Protocol) protocol, supplemented with RPC (Remote Procedure Call) protocol which enables secure calls from a client to a server, is used to monitor communication and identify local objects. In particular the tasks are:

- Establishment of a communication relationship during startup;
- Assignment between local objects and communication objects;

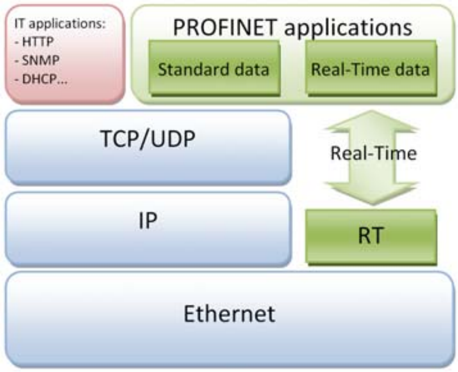



**Fig. 3.** PROFINET I/O communication stack

– Parameter assignment during startup;
– Reading detailed diagnostic information;
– Exchanging general device information;
– Reading/modifying device parameters;
– Loading/reading process-relevant information;
– Reading and changing general communication parameters.

In the presented analyzer solution this kind of communication is used for network structure discovery. This structure may be read from network project and GSD files but in case of real network analysis solution, such project assumption allows for dynamic network structure discovery.

After communication has been established, CR communication channels are used for cyclical data exchange. On this level described analyzer uses mirror port on PROFINET I/O switch dedicated to passive listening of PROFINET I/O communication frames. PROFINET I/O uses Ethernet frames that are identified by their special Ethertype (0×8892) as PROFINET real-time frames. Ethertypes are assigned by IEEE and are a unique criterion for distinguishing from other protocols [5,6].

The FrameID is used to address a specific communication channel between I/O-Controller and I/O-Device (Fig. 4). Ethertype together with FrameID are the two elements of PROFINET I/O protocol which enable a fast selection of RT frames without any additional frame headers. To find the correct communication channel, PROFINET I/O device needs to only decode the Ethertype and FrameID fields.

The RT channels are used for specific functionalities, such as:

– Time synchronization;
– Cyclic data transfer;
– Acyclic transfers;
– Alarms and events handling.

For the cyclic data transfer the protocol provides other hardware dependent channels:

| Dest Addr | Src Addr | VLAN +EtherType (optional) | Ether Type | FrameID | User Data Block | Cycle Counter | Data Block Status | Transfer Status | Frame Control Sum |
|---|---|---|---|---|---|---|---|---|---|
| 6 | 6 | 4 | 2 | 2 | ... | 2 | 1 | 1 | 4 |

**Fig. 4.** PROFINET RT frame structure

– RT class 1 – can be executed with standard switches;
– RT class 2 – requires special switches, communication scheduling is not required;
– RT class 3 – requires special switches and communication scheduling (IRT mode);

The described PROFINET I/O network analyzer allows RT communication analysis on RT1 and RT2 class level. The process data for specific modules is exchanged by means of UserDataBlock field. Cyclic data (process data) is transmitted between provider and consumer without any additional security or acknowledgement frames. The absence of cyclic data is monitored using a configured monitoring time slot. If there are no new values within the set monitoring time slot, error message is passed.

The status of process data is evaluated and signaled using the following fields [6]:

– Cycle Counter: A provider increments the cycle counter in each cycle by 31.25 micro seconds and inserts it, consumer checks the counter and derives timeliness and validity of the data;
– Data Status consists of 4 bits:
– State : 1=Primary, for redundant channel;
– DataValid: 1=Data are valid;
– ProcessState: 1=Process is running;
– ProblemIndicator: 1=No problems exist.
– Transfer Status: must be 0.

That mechanism of time and data status monitoring enables a rapid response to a failure and doesn't put too much load on the system. PROFINET I/O standard improves data transfer in the network by usage a priority index which is compatible with IEEE 802.1Q standard [1].

IEEE 802.1Q standard allows for building Virtual LAN networks which utilize one link for transparent transmission of data coming from different networks. Every virtual network possesses a priority index. That way the network traffic generated by particular application can be transferred in the network with a preferential processing in switching or routing devices. The devices use a special packets tagging (Fig. 5).

The Ethertype field is set to 0x8100 value what indicates IEEE 802.1Q frame. After the field 4 bytes are added:

– 3-bit "Priority" field – the higher value the higher priority;

| | VLAN (optional) | | | |
|---|---|---|---|---|
| EtherType 8100 | **Priority** | 0 | VLAN-ID | original EtherType |
| 2-byte | 3-bit | 1-bit | 12-bit | 2-byte |

**Fig. 5.** VLAN-Tag

– CFI bit (Canonical Format Indicator) – tells about the technology the LAN network is built in. It is “0” for Ethernet and “1” for Token Ring;
– 12-bit VLAN-ID field – response to the virtual network id the frame belongs to;
– 2 bytes for saving the original EtherType value (for further processing).

Real-time data are signed by the standard with the index of 6, what provides the data a priority in relation to other data, e.g. VOIP calls which have the index of 5 [2].

## 3 The PROFINET I/O Analyzer Idea

Presented PROFINET I/O Analyzer enables a user to monitor, diagnose and examine a PROFINET I/O system with a minimum disturbance level on the working industrial network. The application is designed for a standard PC station. The access to the PROFINET I/O network data is provided by means of standard Ethernet interface and “mirroring” capability of network switch. The “mirroring” feature allows for copying network traffic from one port to another. If controller’s port is mirrored, a full image of the data exchange between PROFINET IO devices is provided for our application.

The captured network frames are then decoded and saved to the data base (Fig. 6). Next the gathered data is being processed and analyzed concerning discovery of network devices, the connections between them and network events. The results are presented to a user in form of network diagram and data tables. The tables contain information regarding the captured frames, like: source and destination addresses, type, protocol, timestamp and decoded data block. Additionally, a general description of a specific frame’s role is provided.

Any occurrence of PROFINET IO alarm is detected and indicated both at the diagram and data table. For every detected device user can find a lot of detailed information: device name, device type, vendor name, device and vendor ids, IP and MAC addresses, as well as a list of alarms concerning the particular device.

Every time the capturing process is started a new session is being created. For the sessions selected by user the communication analysis and visualization is performed. This gives the user possibility of information analysis retrieved from many sources (e.g. if there are more controllers in the network). The application is meant to be freeware licensed; therefore the software tools and libraries used to build the application are also freeware.

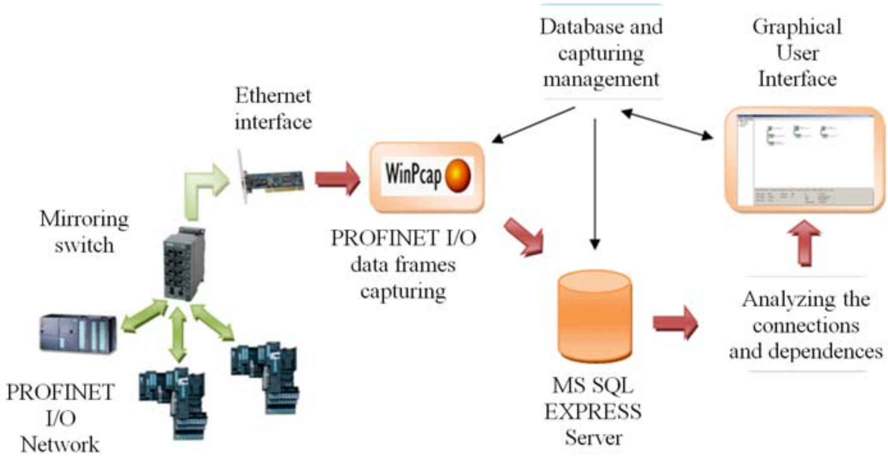



**Fig. 6.** PROFINET I/O analyzer structure

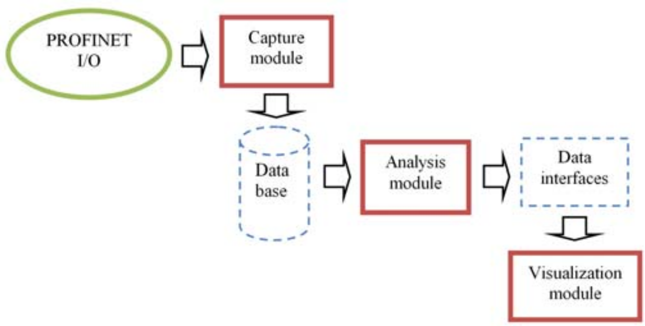



**Fig. 7.** Data flow in PROFINET I/O analyzer modules

Application architecture consists of three functional modules distinguished in the application project:

– Capturing – handling network driver services, data filtering, decoding frames, writing data into the database;
– Analysis – reading network frames from the database and retrieving useful information;
– Visualization – presenting the results by drawing a network diagram and a structure of the captured frames.

The intermediary element between the modules are database and data interfaces. The database is designed to store frames of different types, as well as, their capture timestamp and session (Fig. 7).

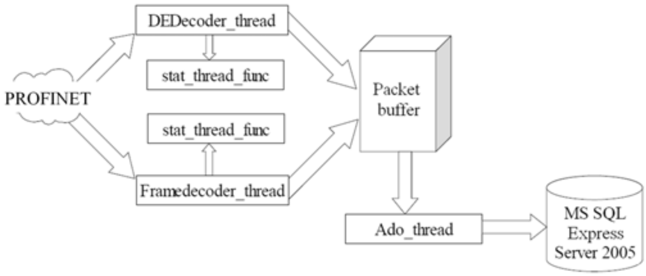



**Fig. 8.** PROFINET I/O analyzer software threads

Due to significant data exchange rate the capture module has to be time optimized to provide high-speed processing and meet time-deterministic requirements. To satisfy these demands the module is based on efficient and proven WinPcap library. Network frames are read by means of Ethernet interface card and are saved in a form of a byte table located in the RAM. Next the frames are initially decoded to filter the data which is useful for analysis. The types of the PROFINET frames are:

– Alarms, Identify, Connect, ARP – for devices, connections and alarms detection;
– DataExchange, Control, Read, Write – process data and parameters.

Before the frames are stored into the database, next decoding phase takes place. All frame's fields are singled out, then timestamp and session number are attached. Decoded frames are written into a memory buffer in form of appropriate data structures. The buffer allows for writing the decoded data into the database independently of capturing and decoding thread.

Additionally capture statistics are collected. The statistics refer to amount of captured and missed frames and are made accessible for the other application modules to control the capturing process. The tasks are processed by means of independent threads (Fig. 8).

As there are relatively a lot of DataExchange frames within overall PROFINET network traffic they are decoded in the separate thread.

Analysis module – the goal of the module is to determine the meaning of captured frames in the scope of PROFINET I/O system. The data contained in frames is stored in form of raw bytes, and therefore unreadable for a user. The raw data has to be "translated" into user friendly information. In result, the module transforms byte strings into texts or numbers according to PROFINET I/O specification rules. The analyzed data is stored in form of appropriate data structures corresponding to system objects, like: devices, connections or alarms. Such information is next used by visualization module.

Frames of different type transfer different piece of information. From ARP frames IP and MAC addresses are retrieved. Alarm frames give us an error code of device error or network event. In case of Connect frames the most important

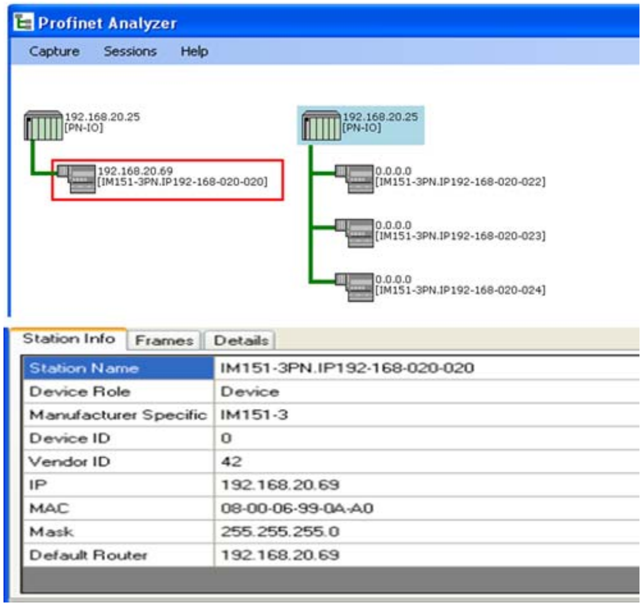



**Fig. 9.** PROFINET I/O analyzer – vis ualization module

pieces of information are MAC addresses of connected devices and established communication channels. The status of logical connections between devices is determined by decoding data status transferred using DataExchange frames. The devices' information and discovery data (name, vendor, network IP settings) is transferred by means of Identify frames. However; Identify frames are normally exchanged only during the startup phase in the PROFINET IO system and therefore Identify request frames are generated by application whenever device discovery is needed. Particular structure and parameters of a device are read from Read, Write and Control frames.

The Visualization module is responsible for presenting the results of analysis and providing interaction between the application and a user (Fig. 9).

The network diagram is drawn in two steps: first, to every network object (device, connection, alarm) stored in the memory in form of data structure a graphical representation is attached, next all elements are displayed in appropriate order. Devices which are sending alarm frames are additionally highlighted. The general view of captured and decoded frames is presented as a table, similarly to existing solutions (Wireshark, Ethereal). Frame type filter is also provided. The GUI allows for control of capturing process and session management. A user can start and stop the capture process and also set capture module parameters, like Ethernet interface or frame filter. The automatically saved sessions can be loaded or removed from the database.

## 4 Conclusions

This article describes the open project realized in Institute of Informatics, Silesian University of Technology with cooperation with University of Applied Sciences Ingolstadt – the network analyzer for PROFINET I/O. Described solution was implemented and tested. The analyzer's idea principles are based on joined operation modes: active industrial network structure scanning and passive listening to a real-time communication. Information about industrial PROFINET I/O static structure is used in the captured frames analyzing process.

PROFINET I/O analyzer consists of three separated parts responsible for: frame capturing, data filtering, decoding and storing to the database, analyze captured network frames and visualization part. This solution can separate real time system's part from off-line system components.

Two different protocol classes are used: TCP/IP stack protocols for network structure discovering and PROFINET I/O RT protocol for dynamic network traffic searching. The idea of low-invasion monitoring network based on switch port mirroring was described as well.

Described project may be the base for the commercial solutions in this area. The purpose for such kind of application is to help to diagnose network problems. This project gives a very good starting point for future scientific research in the area of industrial networks based on Ethernet protocol.

Many implementation problems at given PROFINET I/O devices were found during this project realization. Some incapability problems related with limited protocol function implementation may cause system realization problems. This gives an idea that such tool may be useful not only as a tool for given application startup and maintenance, but it may be helpful for new and existing PROFINET devices testing processes.

## References

1. IEEE Computer Society: 802.1AB Station and Media Access Control Connectivity Discovery. IEEE Computer Society, New York (2005)
2. Sheble, Nicholas: Siemens CBA: Object on the Ethernet, http://www.allbusiness.com/manufacturing/computer-electronic-product-manufacturing/895226-1.html
3. Kleines, H., Detert, S., Drochner, M., Suxdorf, F.: Performance Aspects of PROFINET IO. IEEE Transactions on nuclear science 55(1) (February 2008)
4. PInternational: PROFINET IO Profiles for decentralized periphery, Version 2.2. PROFIBUS Nutzerorganisation e.V., Karlsruhe (2007)
5. PInternational: GSDML Specification for Profinet IO, Version 2.16. PROFIBUS Nutzerorganisation e.V., Karlsruhe (2008)
6. PInternational: PROFINET IO Application Layer protocol for decentralized periphery and distributed automation Version 2.2. PROFIBUS Nutzerorganisation e.V., Karlsruhe (2007)
7. Popp, M., Weber, K.: The Rapid Way to PROFINET. PROFIBUS Nutzerorganisation e.V., Karlsruhe (2004)
8. Siemens AG: PROFINET Technology and Application. Siemens information materials, Karlsruhe (2005)

# Dual Bus as a Method for Data Interchange Transaction Acceleration in Distributed Real Time Systems

Andrzej Kwiecień[1] and Marcin Sidzina[2]

[1] Silesian University of Technology, Institute of Informatics
andrzej.kwiecen@polsl.pl

[2] University of Bielsko-Biala, Department of Mechanical Engineering Fundamentals
msidzina@ath.bielsko.pl

**Abstract.** The chapter presents considerations on data interchange acceleration method in distributed control systems by the use of dual data transmission line. The method is based on the construction of a system based on two buses: one with separated master bus and the other one with separated slave bus.

## 1 Introduction

Distributed systems based on Master-Slave line access method are characterized by the fact that Slave devices are "subordinated" and can never establish communication in a network system. It follows that the whole interchange scenario is managed by the Master station. Master station is the only one that has the possibility to establish network communication. It is Master station that prompts for replies from other stations, sends data-writing requests to the address space of a Slave station. The possibility of communication being triggered by a Slave station is practically seldom discussed in literature. If such a discussion is taken up it is only where a possibility of an IT node failure or a facility failure is considered, yet outside an IT system. Execution of special procedures prepared for such events is limited by a number of conditions to be fulfilled. However, the procedures are used only in critical situations e.g. when an IT system is heading for collapse.

Network protocol with an access to a Master-Slave line, such as e.g. MODBUS, was defined for a single communication bus. Device coprocessor operating in MODBUS network does not serve two buses, as a standard. In order that the network reliability and safety improvement could be considered the system nodes must be equipped with additional network interfaces (communication coprocessors) operating in MODBUS system. The publication [4] presents such a MODBUS-based redundant system. The following redundancy options were taken into consideration at that point:

– redundancy of SLAVE station coprocessors;
– redundancy of MASTER station coprocessors;
– redundancy of both MASTER station and SLAVE station coprocessors;
– redundancy of the bus.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 252–263, 2009.

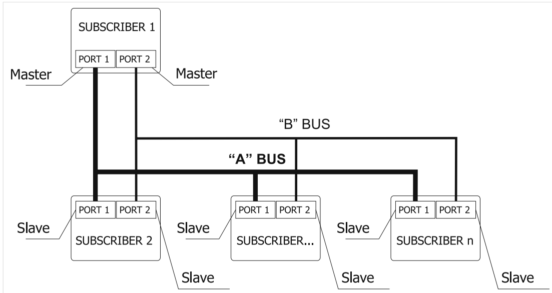



**Fig. 1.** Bus redundancy examples with active MASTER and SLAVE stations marked

Based on the above cases test stations were configured for testing. Two base configurations [4] provided basis for the systems to be tested.

**Configuration 1**, which employs a solution with two independent network buses (Fig. 1). Each of the buses operates independently. Each bus incorporates one MASTER station and $n$ SLAVE stations. MASTER stations were programmed so as to implement the same interchange scenario.

In such a case each of the buses should be viewed as a separate communication system and each "query/command-response" transaction execution time for each bus is as follows:

$$T_{\mathrm{W_{ZPO}}} = T_{\mathrm{ZM}} + T_{\mathrm{PM}} + 2\ \left(T_{\mathrm{PR}} + T_{\mathrm{TR}} + T_{\mathrm{DT}} + T_{\mathrm{AR}}\right) + T_{\mathrm{AS}} \tag{1}$$

where:

$T_{\mathrm{W_{ZPO}}}$ – "query/command-response" interchange execution time;
$T_{\mathrm{ZM}}$ – time measured from interchange request during $T_{\mathrm{AP}}$ cycle stage to commencement of $T_{\mathrm{K}}$ stage;
$T_{\mathrm{AS}}$ – duration of SLAVE station cycle;
$T_{\mathrm{PM}}$ – time from information decoding by the MASTER station coprocessor to the moment the information is accepted and used in MASTER station application;
$T_{\mathrm{PR}}$ – frame preparation time;
$T_{\mathrm{TR}}$ – frame transmission time;
$T_{\mathrm{DR}}$ – frame detection time;
$T_{\mathrm{AR}}$ – frame analysis time.

For broadcast interchange it was assumed that:

$$T_{\mathrm{W_R}} = T_{\mathrm{ZM}} + T_{\mathrm{PR}} + T_{\mathrm{TR}} \tag{2}$$

$T_{\mathrm{W_R}}$ – broadcast interchange execution time;
$T_{\mathrm{ZM}}$ – time from interchange request during automaton cycle stage to the commencement of communication stage.

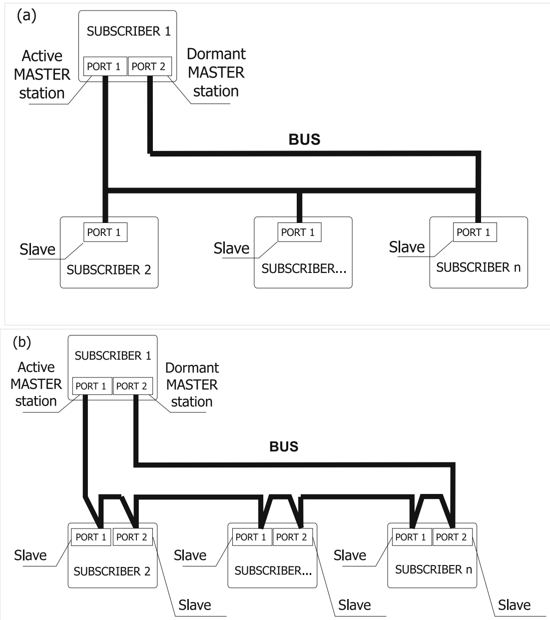



**Fig. 2.** Bus redundancy examples with active MASTER and SLAVE stations marked

**Configuration 2**, presents a redundant system based on one network bus and two MASTER stations one of which is "dormant". System bus is a "pseudo-ring" (Fig. 2).

The configuration assumptions provide for the communication system operating on "one Master many Slaves" rule. In case of a failure resulting from a break in the bus, the connection allows the activation of the "dormant" Master station to enable communication with the "cut-off" Slave stations. The publication [4] determines an algorithm (method) for the Master station which enables to determine the occurrence of an event of a break in the transmission line.

Data interchange time model for configuration 2 is the same as the one for configuration 1. An important aspect of the configuration presented is the determination of response time related to the activation of the "dormant Master station" combined with detection of the bus failure point. It was defined [4] as the following relation:

$$T_{\mathrm{M2}} = 2\ (T_{\mathrm{PR}} + T_{\mathrm{TR}} + T_{\mathrm{TIMEOUT}}) + 2\ (T_{\mathrm{PR}} + T_{\mathrm{TR}} + T_{\mathrm{DT}} + T_{\mathrm{AR}}) + 3\ (T_{\mathrm{AS}}) \tag{3}$$

where:

$T_{\mathrm{M2}}$ – activation time of the other MASTER station with bus failure point detection;
$T_{\mathrm{TIMEOUT}}$ – time of waiting for a response from the SLAVE station;
$T_{\mathrm{A}}$ – duration of automaton cycle.

The presented examples of redundancy show how to improve network reliability by doubling coprocessors or communication buses. Taking into account the time analysis of data interchange transactions in the configurations discussed it can be noted that those configurations are not conducive to system performance improvement. The bus redundancy has no direct impact on increase in data interchange frequency, except for a situation of a break in the network in configuration no. 2. Natural segmentation of the buses, in turn, caused by e.g. a failure, consisting in isolation of two, in a sense, independent buses, will cause simultaneous shortening of the network cycle duration if assuming moderately balanced ratios between two divided network parts. It must be noted however that such a state is not natural for the configuration in question and it occurs only in case of a break in the transmission line.

One of the main parameters in distributed real-time systems, in addition to the data interchange transaction security, is the network cycle duration. It is said for the central interchange scenario networks that the network cycle duration is the time required by a MASTER station to perform all data interchange transactions. For both configurations the time necessary to carry out the interchange scenario can be described with the following relations:

– For **Configuration 1**

$$T_{\mathrm{CS}_{\mathrm{busA}}} = \sum T_{\mathrm{W}_{\mathrm{busA}}} \quad \text{and} \quad T_{\mathrm{CS}_{\mathrm{busB}}} = \sum T_{\mathrm{W}_{\mathrm{busB}}} \tag{4}$$

where:

$T_{\mathrm{CS}_{\mathrm{busA}}}$ – network cycle duration on “A” bus;
$\sum T_{\mathrm{W}_{\mathrm{busA}}}$ – sum of durations of all transactions listed in the interchange scenario on “A” bus;
$T_{\mathrm{CS}_{\mathrm{busB}}}$ – network cycle duration on “B” bus;
$\sum T_{\mathrm{W}_{\mathrm{busB}}}$ – sum of durations of all transactions listed in the interchange scenario on “B” bus.

– For **Configuration 2**

$$T_{\mathrm{CS}} = \sum T_{\mathrm{W}} \quad \text{or} \quad T_{\mathrm{CS}} = \sum T_{\mathrm{w\_czA}} + \sum T_{\mathrm{w\_czB}} \tag{5}$$

where:

$T_{\mathrm{CS}}$ – network cycle duration;
$\sum T_{\mathrm{W}}$ – sum of durations of all transactions listed in the interchange scenario;
$\sum T_{\mathrm{w\_czA}}$ – sum of durations of all transactions carried out on the broken bus in “A” part;
$\sum T_{\mathrm{w\_czB}}$ – sum of durations of all transactions listed in the interchange scenario on “B” bus.

## 2 Use of Doubled Communication Bus to Dynamically Shorten the Network Cycle Duration

The previous chapter briefly describes two architectures of systems tested which use computer network with applied redundancy. The tests presented [4] were aimed at determination of delay values in the real time system, resulting from the use of redundant structures. The considerations presented in this chapter, in turn, are the possibilities to use redundant network structures in the existing and operating systems and also is newly designed ones to improve time parameters (e.g. a parameter of *useful throughput* $P_{\mathrm{U}}$ *or useful efficiency* $\eta'_{\mathrm{U}}$) of computer networks. It is a very important issue as the network efficiency has an impact on such parameters as *limit time* ($T_{\mathrm{GR}}$) of the real time system but also quality e.g. that of the adjustment performed by a programme residing in a computer unit being a system node. As it was demonstrated in [1], [6] the said parameters have an impact on both network cycle duration and node cycle duration. Therefore, it seems interesting to use the backup communication bus to carry out data interchange in a network when the basic bus is overloaded which poses a threat of e.g. the critical parameters being exceeded.

Such reasoning inclined the authors to propose a new transmission method based on the existing communication protocols and certain assumption. The first of them is related to the selection of the basic communication protocol layer which is to provide a basis for the operation of the method proposed. The fundamental assumption of the solution proposed is the implementability thereof based on both the existing hardware (*PLC – Programmable Logic Controllers*) together with its programming tools and the existing communication protocols being a core of the method. It is of importance from the research point of view as it allows to develop a testing station without the need to bear significant costs related to construction works the production of a network controller prototype along with software. For, as a matter of fact, the solution proposed offers new communication protocol based on fundaments of the existing protocols. The tests proposed are aimed at full time analysis of the new protocol and determination of its capabilities and usefulness in designing new real-time industrial systems. Only after the results obtained are accepted work on a new network controller can be launched.

Thus, it was MODBUS RTU protocol of the Master-Slave protocol family that was chosen as a base protocol. With the hardware and software available there were no significant problems relate to development and implementation of the programme layer residing in the memory of the central unit serving the proposed operating modes of the new network. A number of programme procedures were created to enable practical implementation of the method proposed.

In order to shorten the network cycle duration for MODBUS network the interchange scenario ([1]) needs to be designed properly. Where it is not possible to shorten the cycles by improving scenario parameters it can be divided into two shorter ones, performed in two different MASTER stations, operating on two communication buses. This commonly known solutions, is defined as *network segmentation.* It enables to shorten the network cycle duration but does

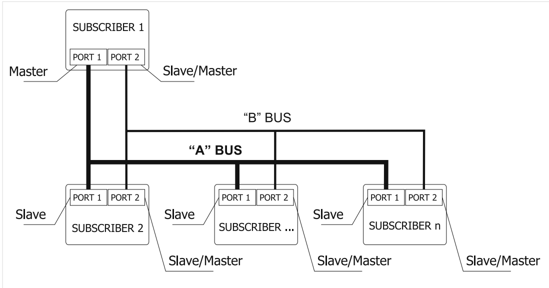



**Fig. 3.** Sample system with the possibility to activate any subscriber as a MASTER station on B bus

not maintain the reliability provided by the redundant structure. Whereas the protocol proposed additionally preserves the improved reliability resulting from the redundancy of the transmission medium.

The proposed protocol was developed for the architecture modelled and presented in **Configuration 1**. The basic difference between **Configuration 1** and the method presented is that according to the below considerations the buses are divided into a "superior" bus (A) and a "subordinated" bus (B). The "A" bus operates on the basis of the basic MASTER-SLAVE data interchange model with a permanently assigned MASTER station and SLAVE stations. All equipment items connected to the "B" bus operate in driven state as SLAVE units. Once the predefined circumstances are in place the MASTER station of the "A" bus sends a control frame to the selected SLAVE station commanding it to perform the MASTER function on "B" bus however. Thus, the appointed SLAVE station still remains a SLAVE type subscriber on "A" bus while becoming a MASTER station on "B" bus at the same time.

Two operating modes have been developed:

– **Mode 1** – the selected station on "B" bus is a MASTER station for the period when it perform one dedicated triggered interchange with any SLAVE station of the B bus. The configuration process will decide on which stations the triggered interchanges will be possible with, while during normal operation the SLAVE station address of the B bus is placed in the control frame. After completion of the triggered transaction the control over the B bus is passed over to the initiating station (Master station of the A bus).
– **Mode 2** – the selected MASTER station remains a MASTER station and carries out its own cycle interchange scenario until it is called off by the initiating station (MASTER station of the A bus). The call-off can be done through the performance by the B bus MASTER station of a cycle interchange with the MASTER station of the A bus during which a suitable control frame is sent indicating the change of the control source.

Thus, in any case the MASTER station of the A bus is somehow a distributor (manager) of the messages on the B bus (Fig. 3).

An additional function of the proposed solution is the conventional emergency mode where B bus becomes the major bus in case there is no communication with the use of A bus.

## 3 Basic Time Relation Model Present in the Proposed Network Configuration

As we move now to the preliminary time analysis of the protocol operation, the following basic data interchange transactions can be distinguished in the protocol:

- **Case 1**
  The analysed data interchange transaction system refers to a periodically implemented scenario of interchanges on A bus, if the B bus is "dormant". Execution time of query/command-response transaction and broadcast transactions is identical with relations (1) and (2) and the network cycle duration is equal to the interchange scenario performance time.

$$T_{\mathrm{CS\_busA}} = \sum T_{\mathrm{W\_busA}} \quad \text{and} \quad T_{\mathrm{CS\_busB}} = NULL \tag{6}$$

where:
$T_{\mathrm{CS\_busA}}$ – network cycle duration on A bus;
$T_{\mathrm{CS\_busB}}$ – network cycle duration on B bus;
$\sum T_{\mathrm{w\_czA}}$ – sum of durations of all interchanges in the Master station scenario on A bus.

*Example 1.* A network was configured composed of four network nodes. The Master station on A bus serves the interchange scenario performing periodically the following data interchange transactions[1]:

| Pos. | Type of periodical data interchange transaction | Max. duration of interchange transaction in ms |
|---|---|---|
| $T_w1$ | Readout of ten words from Slave No. 1 station | 33.5 |
| $T_w2$ | Readout of ten words from Slave No. 2 station | 33.5 |
| $T_w3$ | Writing of 16 bits to Slave No. 3 station | 29.2 |
| $T_w4$ | Writing of 16 bits to Slave No. 1 station | 29.2 |

Anticipated max. duration of data interchange transaction is:

$$T_{\mathrm{CS\_busA}} = \sum T_{\mathrm{W\_busA}}$$
$$\sum T_{\mathrm{W\_busA}} = T_w1 + T_w2 + T_w3 + T_w4$$
$$\sum T_{\mathrm{W\_busA}} = 125.4\,\mathrm{ms}$$

[1] Values obtained empirically for Modbus network. Results included in papers [4], [6].

– **Case 2**
Where it is necessary to shorten the network cycle duration or increase the message exchange between certain subscribers, the interchange scenario is extended by the time of necessary to perform a command to activate one of SLAVE stations as a MASTER station on B bus. It should be assumed that the command is an aperiodic transaction which breaks the periodical performance of a transaction on A bus. Performance of the command has a direct impact on increase in A bus network cycle performance time.

$$T_{\mathrm{CS\_busA}} = \left(\sum T_{\mathrm{W\_busA}} + T_{\mathrm{W\_rozkaz}}\right) - \sum T_{\mathrm{W\_busB}} \tag{7}$$

where:
$[\sum T_{\mathrm{W\_busB}}]$
$T_{\mathrm{W\_rozkaz}}$ – B bus activation command;
$\sum T_{\mathrm{W\_busB}}$ – Sum of durations of all interchanges in the Master station scenario on B bus.

The B bus activation time is defined by the following relation ([8]):

$$\begin{aligned} T_{\mathrm{start\_B}} = {} & \left(T_{\mathrm{AZ\_rozkaz}} + T_{\mathrm{PR}} + T_{\mathrm{TR}}\right) + \\ & \left(T_{\mathrm{DR}} + T_{\mathrm{AR}} + T_{\mathrm{AS}} + T_{\mathrm{AW}} + T_{\mathrm{A}} + T_{\mathrm{AK}}\right) + \\ & \left(T_{\mathrm{PR\_busB}} + T_{\mathrm{TR\_busB}} + T_{\mathrm{DR\_busB}} + T_{\mathrm{AR\_busB}} + T_{\mathrm{AM}}\right) \end{aligned} \tag{8}$$

where:

$T_{\mathrm{start\_B}}$ – B bus activation time;
$T_{\mathrm{AZ\_rozkaz}}$ – Time measured from the occurrence of the activation request of the B bus in the central unit to the transfer of data to the MASTER coprocessor on A bus;
$T_{\mathrm{AS}}$ – Time of waiting for completion of the SLAVE station automaton cycle;
$T_{\mathrm{AW}}$ – Time measured from the acceptance of the command to activate the other interface as a MASTER station on B bus to the performance of the activation command;
$T_{\mathrm{AK}}$ – Coprocessor activation time;
$T_{\mathrm{PR\_busB}}$ – Preparation time for MASTER station activation confirmation frame on B bus;
$T_{\mathrm{TR\_busB}}$ – Transmission time for MASTER station activation confirmation frame on B bus;
$T_{\mathrm{DR\_busB}}$ – Detection time for MASTER station activation confirmation frame on B bus;
$T_{\mathrm{AR\_busB}}$ – Analysis time for MASTER station activation confirmation frame on B bus;
$T_{\mathrm{AM}}$ – Time between data readout from the SLAVE station coprocessor to the data transmission to the central unit operating with the MASTER station on A bus.

Both relations (7) and (8) define the theoretically maximum time by which the network cycle performance time on A bus will be extended. All parameters included in the relations (7) and (8) are either known values (catalogue data of the equipment manufacturer) or they can be easily determined in an experiment.

*Example 2.* A network was configured composed of four network nodes (Fig. 4). The Master station on A bus serves the interchange scenario performing periodically the following data interchange transactions[2]:

| Pos. | Type of periodical data interchange transaction | Max. duration of interchange transaction in ms |
|---|---|---|
| $T_w1$ | Readout of ten words from Slave No. 1 station | 33.5 |
| $T_w2$ | Readout of ten words from Slave No. 2 station | 33.5 |
| $T_w3$ | Writing of 16 bits to Slave No. 3 station | 29.2 |
| $T_w4$ | Writing of 16 bits to Slave No. 1 station | 29.2 |

The interchange scenario includes two aperiodic data interchange transactions:

| Pos. | Type of periodical data interchange transaction | Max. duration of interchange transaction in ms |
|---|---|---|
| $T_w5$ | Readout of ten words from Slave No. 1 station | 29.2 |
| $T_w6$ | Readout of ten words from Slave No. 2 station | 29.2 |

After completion of several basic cycles on A bus, $T_w5$ command is performed which activates the Master station on B bus. The B bus takes over $T_w1$ and $T_w4$ data interchange transactions while $T_w2$ and $T_w3$ transactions are carried out on A bus.

The anticipated max. data interchange duration on A and B buses is:

- **Stage 1:** periodical transactions are performed only by A bus
  - A bus; performed periodical transactions:

$$T_{\mathrm{CS\_busA}} = \sum T_{\mathrm{W\_busA}}$$

$$\sum T_{\mathrm{W\_busA}} = T_w1 + T_w2 + T_w3 + T_w4$$

$$\sum T_{\mathrm{W\_busA}} = 125.4\,\mathrm{ms}$$

  - B bus; no transactions:

$$T_{\mathrm{CS\_busB}} = NULL$$

- **Stage 2:** A bus performs periodical transactions + one aperiodic transaction; part of the scenario is passed to B bus,

[2] Values obtained empirically for Modbus network. Results included in papers [4], [6].

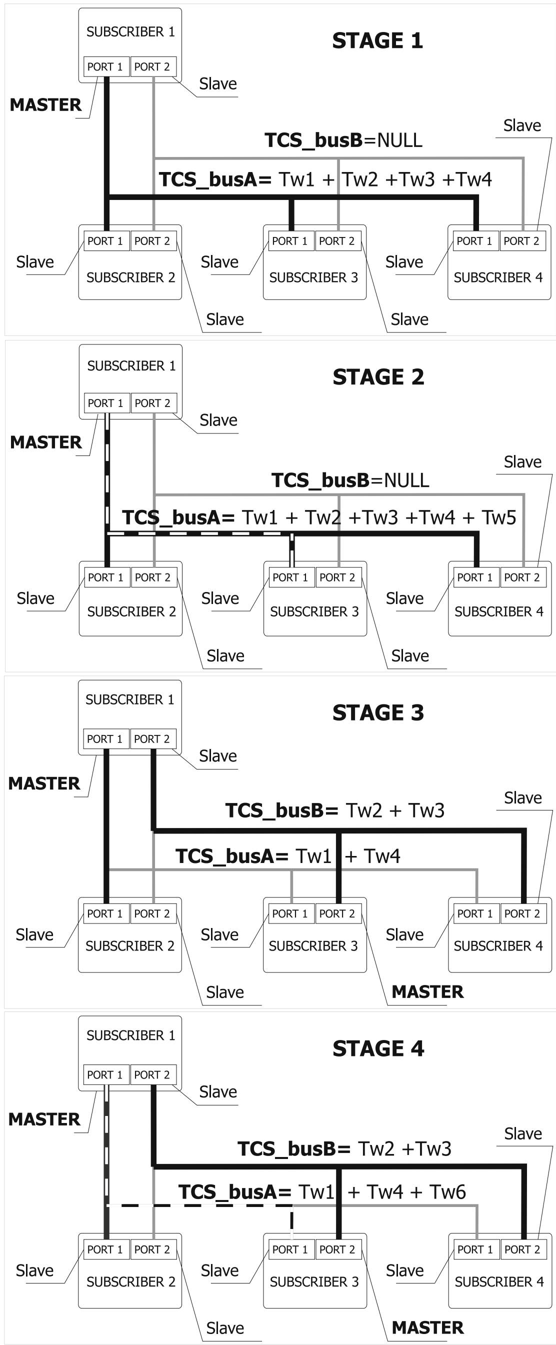



**Fig. 4.** Sample system with the possibility to activate and deactivate a subscriber as a MASTER station on the B bus

* A bus; performed periodical transactions:

$$\sum T_{\mathrm{W\_busA}} = T_{\mathrm{w}}1 + T_{\mathrm{w}}2 + T_{\mathrm{w}}3 + T_{\mathrm{w}}4 + T_{\mathrm{w}}5$$
$$\sum T_{\mathrm{W\_busA}} = 154.6\,\mathrm{ms}$$

* B bus; no transactions:

$$T_{\mathrm{CS\_busB}} = NULL$$

- **Stage 3:** A bus and active B bus perform the interchange scenario
  * A bus; performed periodical transactions:

$$\sum T_{\mathrm{W\_busA}} = T_{\mathrm{w}}1 + T_{\mathrm{w}}4$$
$$\sum T_{\mathrm{W\_busA}} = 62.7\,\mathrm{ms}$$

  * B bus; performed periodical transactions:

$$\sum T_{\mathrm{W\_busB}} = T_{\mathrm{w}}2 + T_{\mathrm{w}}3$$
$$\sum T_{\mathrm{W\_busB}} = 62.7\,\mathrm{ms}$$

- **Stage 4:** Master station sends a command from A bus to shutdown the Master station on the B bus. Once the command is performed the network returns to the state of Stage 1.
  * A bus; performed periodical transactions:

$$\sum T_{\mathrm{W\_busA}} = T_{\mathrm{w}}1 + T_{\mathrm{w}}4 + T_{\mathrm{w}}6$$
$$\sum T_{\mathrm{W\_busA}} = 91.9\,\mathrm{ms}$$

  * B bus; performed periodical transactions:

$$\sum T_{\mathrm{W\_busB}} = T_{\mathrm{w}}2 + T_{\mathrm{w}}3$$
$$\sum T_{\mathrm{W\_busB}} = 62.7\,\mathrm{ms}$$

The above calculations based on empirical investigation carried out for Modbus network show that the use of the subject system will significantly shorten the duration of the global network cycle. The cycle duration was reduced by 50% for the case presented above.

## 4 Summary

This paper presents a proposal of a new communication protocol which can be used in designing industrial IT real time systems. It was created on the basis of examined redundant structures [4] with the use of the existing protocol of MODBUS RTU. The reason for using that protocol was only to construct a model of a new protocol so as to examine it and determine the practical usefulness of the proposed solution for designing industrial applications. The partial results of practical verification of the concept obtained so far and the results of the theoretical considerations presented in Sect. 3 are encouraging and stimulate further tests and examinations which, as mentioned before, will provide a basis to design a prototype controller with own built-in software to perform a new protocol algorithm. An intention of the authors is to design a controller which would be able to operate with the existing hardware solutions. Additional asset of the proposed solution is the fact that it can be employed in the existing systems after installation of additional special programme procedures for the existing application software. Currently, work is conducted on software for the presented communication system to experimentally verify the features of increased reliability and efficiency. The analysis of basic time assumptions allows to state that the method will be a useful tool enabling to expand operation of some communication systems of the "monomaster" architecture.

## References

1. Kwiecień, A.: Analiza przepływu informacji w komputerowych sieciach przemysłowych (in Polish). Studia Informatica, Gliwice (2002)
2. Kwiecien, A., Sidzina, M.: The programmable logic controller PLC work cycle time influence on exchange data time in industrial networks (in Polish). In: AUTOMATION 2004, Warszawa, pp. 593–601 (2004)
3. Sidzina, M.: Badanie przepływu informacji z sieci MASTER-SLAVE do sieci TOKEN-RING z potwierdzeniem lub bez (in Polish). Zeszyty Naukowe. Budowa i Eksploatacja Maszyn, 128–135. Akademia Techniczno-Humanistyczna, Bielsko-Biała (2004)
4. Kwiecień, A., Stój, J., Sidzina, M.: Analiza wybranych architektur redundantnych z zastosowaniem sieci MODBUS/RTU (in Polish). Sieci komputerowe. Praca zbiorowa Tom 2: Aplikacje i zastosowania, 359–367. WKŁ, Warszawa (2007)
5. Kwiecień, A., Sidzina, M., Stój, J.: Synchronizacja sterowników PLC w systemach z redundancją jednostki centralnej (in Polish). Systemy Czasu Rzeczywistego: metody i zastosowania. Praca zbiorowa, 167–177. WKŁ, Warszawa (2007)
6. Sidzina, M.: Dynamiczne modyfikacje programu aplikacji sterownika swobodnie programowalnego celem zwiększenia częstości wymian komunikatów w przemysłowych systemach rozproszonych czasu rzeczywistego. Rozprawa doktorska, Politechnika Śląska Wydział Automatyki Elektroniki i Informatyki, Gliwice (2008)

# Hierarchical Petri Net for the CPDev Virtual Machine with Communications

Dariusz Rzońca and Bartosz Trybus

Rzeszow University of Technology, ul. W. Pola 2, 35-959 Rzeszów, Poland
drzonca@prz-rzeszow.pl, btrybus@prz-rzeszow.pl

**Abstract.** Hierarchical Coloured Petri nets are used to model interaction of CPDev virtual machine with communication subsystem of a small distributed control and measurement system. The machine executes control programs and implements communications. Both vertical and horizontal communication are modelled, as shown by the example of the distributed system with new SMC programmable controller. The model can be used to analyse correlation between the machine cycle time and communication load and to determine optimal parameters of the system.

## 1 Introduction

The CPDev environment (Control Program Developer) [1] is a collection of tools for programming small control devices according to the IEC 61131-3 standard. The main goal of the standard is to increase quality of control software [2]. By defining dedicated programming languages and implementation procedures it frees designers from using general purpose languages like C, focusing instead on application of control algorithms.

The main component of CPDev environment is a compiler which transforms programs written in ST or IL, two of five IEC 61131-3 languages, into an universal executable code. Specification of the universal code has been prepared in such a way, that the resulting binary code can be executed on different target platforms, so both on small microcontrollers and on larger microprocessors, via target-specific virtual machines. The machines operate as interpreters of the universal code. The basic machine is written in industry standard C, so implementation on different platforms does not pose a major problem. Hardware-based machine involving FPGA technology is under development.

Correctness and efficiency are important factors of control software. Formal models used in the early stages of the development can be very helpful in this area. This paper describes Hierarchical Timed Coloured Petri net (HTCP-net) model [3] of the CPDev target platform software. One of the advantages of HTCP-nets is possibility to estimate various performance parameters, thus identify potential bottlenecks and achieve better performance. A consistent description of different aspects of a system is also an important feature of such models. The model described in this paper contains basic operation the CPDev virtual

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 264–271, 2009.


**Fig. 1.** SMC-based mini-DCS system at Automaticon 2008 fair

machine and focuses primarily on coordination of the machine task cycle with communication subsystem. This includes both interaction with host PC (vertical communication) as well as interaction with slave input/output modules and other field devices (horizontal). The two aspects play important role in the first application of the machine in a small distributed control and measurement system (mini-DCS) involving new SMC controller introduced recently by LUMEL Zielona Góra (Fig. 1). The controller does not have inputs or outputs of its own, but communicates instead with dedicated I/O modules via the Modbus protocol [4]. Modbus is also used for PC–SMC transactions. A part of the SMC Petri net model related to vertical communication has been presented in [5,6].

## 2 Hierarchical Model of the Virtual Machine

The basic part of the CPDev virtual machine model is shown in Fig. 2. The HTCP-net represents activities performed sequentially by the machine during a task cycle. Those involve reading and writing inputs and outputs, executing a task and waiting for the end of the task cycle. The token passed between the places holds an extra integer variable (`ct`). It is used to collect delays introduced by succeeding activities. For example, the arc expression `(x,ct+read_time)` below the transition *Read inputs* is used to increase the total delay by the value *read_ time*, which can be set as necessary. To make the cycle constant, the total delay is used to calculate the time left until the cycle ends. The spare time is modelled with a time stamp expression `@+(task_cycle-ct)`, near the transition *Wait for cycle end*, which prohibits a token's use until the model time reaches the appropriate value. It has been assumed here, that *the communications take place during the spare time period of the task cycle.*

Hierarchical properties of the HTCP-nets which have been used in the CPDev virtual machine model, are:

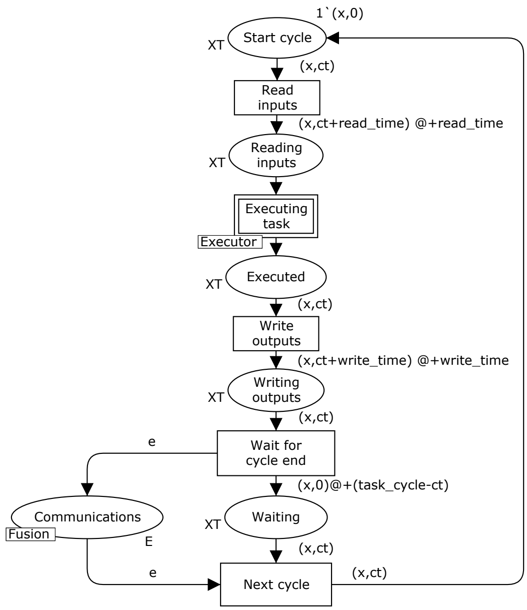



**Fig. 2.** HTCP-net of the virtual machine cycle

– substitution transitions connecting superpages and subpages via input/output ports
– fusion place set to distribute token flow between net pages.

The *substitution transition* concept has been used in the model to provide more and less detailed levels of system description. This can be observed in the transition *Execute task*, which can be considered as a "gate" to a lower level description of task execution. In the discussed model, this evolves into the subpage net shown in Fig. 3.

For small-scale appliances as considered here, only one task is allowed. A task in the CPDev environment is defined by a set of programs executed sequentially. To begin a task, a token is passed from the superpage via the input port *Task execution* (Fig. 3). *Counter* is used to get subsequent programs. Each program is run within a specified amount of time, so the total cycle delay is accordingly adjusted by the function `addExecTime` at the transition *Run program*. A token with no execution time (`pt=0`) indicates that there are no more programs to run. Consequently, the output port *Task finished* passes the token back to the superpage of Fig. 2.

The place *Communications* on Fig. 2 is a *fusion place*, belonging to a *fusion set*, which is another aspect of HTCP-nets. Generally speaking, a fusion set provides multiple representations of the same place, thus it can be used on different pages. A token `e` is passed to the fusion place *Communications* at the beginning of the spare time, what activates communication subsystem (described

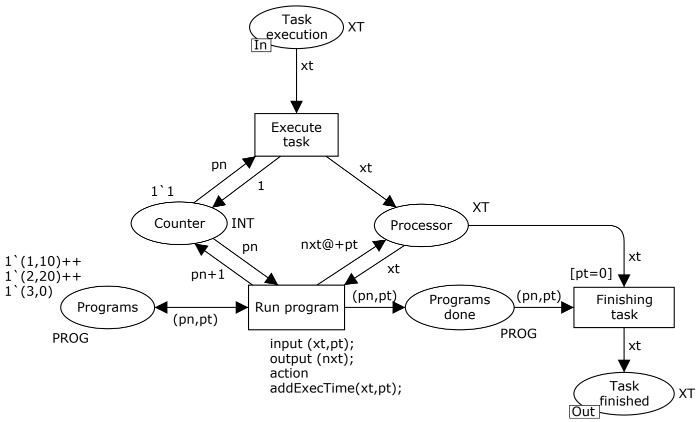



**Fig. 3.** HTCP-net of the task execution

further). When the cycle timeout is reached, the token is grabbed back, so the communications will pause until the next spare period.

## 3 Communications between PC and SMC

HTCP-net model of communication between PC and SMC is shown in Fig. 4. Left part of the net corresponds to a PC running CPSim (CPDev's on-line testing tool) or SCADA program. Transitions *Net PC 1* and *Net PC 2* (center) represent transmission link between PC and SMC. They are substitution transitions introducing lower level subnets (described later). Right part of the figure models SMC behaviour.

Initial marking of the place *Enable to send* enables the transition *Sending*. When the transition fires, a message to be sent is chosen from the tokens available in *Messages*. The selected message is put into *PC outgoing buffer* as well as another token into *Waiting for response*. The arc inscription `@+600` from *Sending* to *Waiting for response* delays this token, so the transition *Timeout* will be active after the selected time. However, the inscription at the next arc allows the transition *PC processing* to consume the token earlier, immediately after receiving response into *PC incoming buffer*. A message represented by a token in *PC outgoing buffer* is transmitted through *Net PC 1* and put into *SMC incoming buffer*. The message waits there, until the SMC virtual machine is ready to process it, that means until the spare time period is reached at the end of the cycle. A token at the fusion place *Communications* enables *SMC processing*, so the message is handled and the response is put in *SMC outgoing buffer*. The response is transmitted back to the PC through *Net PC 2* and put in *PC incoming buffer*. This allows the transition *PC processing* to fire. A token in the place *Enable to send* enables *Sending*, so next message can be transmitted.

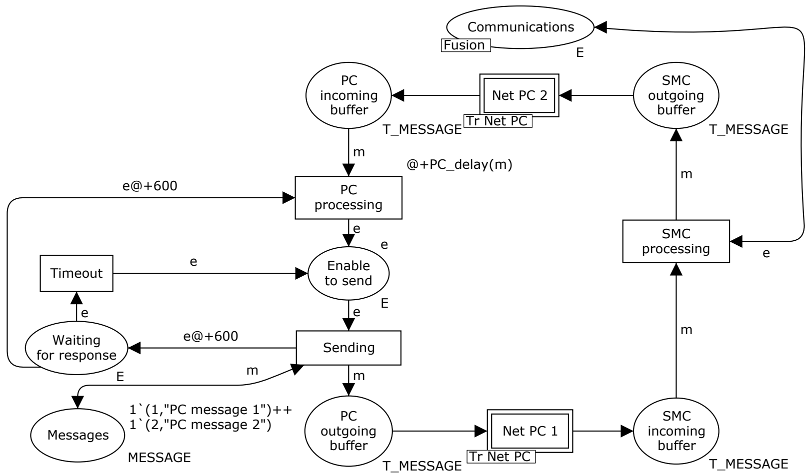



**Fig. 4.** PC–SMC communications as HTCP-net

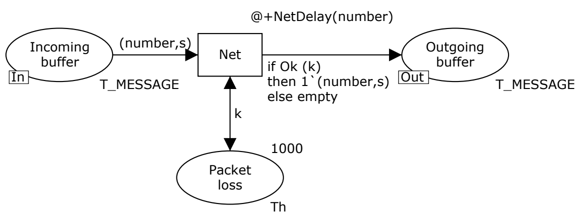



**Fig. 5.** Subnet for PC–SMC transmission

As mentioned, *Net PC 1* and *Net PC 2* are substitution transitions and introduce subpages. In our case they are two separate instances of HTCP-net presented in Fig. 5. However, by the hierarchy, other models of transmission net can be applied if necessary, without a change of the main superpage. The *Net PC* subpage is connected to the superpage with two places: the input port *Incoming buffer* and the output port *Outgoing buffer*. Tokens in *Incoming buffer* represent packets which are ready to be transmitted. The transition *Net* delays the tokens according to the function *NetDelay*. This way, a transmission delay is taken into account. A packet can be also lost during transmission, with a specified probability, thanks to the conditional inscription at the arc from *Net* to *Outgoing buffer*.

## 4 Communications between SMC and Slave Devices

A preliminary version of the HTCP-net model for the SMC communications subsystem has been described in [5] and [6]. Formal techniques helped to

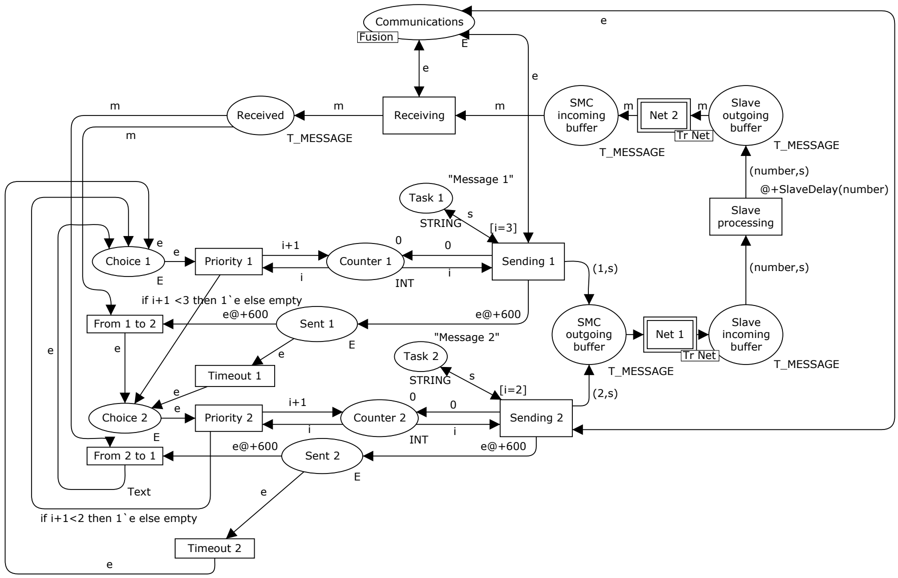



**Fig. 6.** HTCP-net model of the SMC communication subsystem

overcome potential problems like bottleneck or blockage at the design stage. The previous model has been slightly changed to fit into the hierarchical structure as a subpage of the main model. The current version is shown in Fig. 6.

The subsystem allows for declaring multiple so called *communication tasks*. Every task implements a single master-slave transaction with a particular slave device (question–answer or command–acknowledgement). For simplification, just two communication tasks are presented in Fig. 6. The tasks are executed sequentially, according to their *priorities*. In the example net, a token at the place *Choice 1* means that the priority of the task number 1 will be checked. The lower number specified in the guard expression in square brackets at the transitions *Sending 1* and *Sending 2*, the higher priority the task has. The places *Counter 1*, *Counter 2* act as counters. In the example, the token in *Counter 1* has a value of 0. When the transition *Priority 1* is fired, this value will be increased by 1. The arc from *Priority 1* to *Choice 2* has a conditional inscription, so the token will be put in *Choice 2* if the value of *Counter 1* is too low to fire *Sending 1*. This means, that the priority of the task is too low, so the priority of next task will be checked. As it is seen, every attempt to execute a task increases the counter. Eventually, one task will be chosen. If a token in the fusion place *Communications* is present, the selected task can be executed. In the example, after several steps the transition *Sending 2* will be fired, a message put into *SMC outgoing buffer*, *Counter 2* zeroed, and a token put into the place *Sent 2*. The inscription at the arc between *Sending 2* and *Sent 2* increases a timestamp of the token to model a timeout. The transition *Timeout 2* will be active after the time specified. The inscription at the arc from *Sending 2* to *From 2 to 1* allows firing

*From 2 to 1* before the timeout, as soon as a reply appears in the place *Received*. If a packet is lost then the response will never be received, so *Timeout 2* will fire after the specified timeout value. A packet put into *SMC outgoing buffer* will be transmitted through the net. The transitions *Net 1* and *Net 2* are substitution transitions introducing subnets, similarly as in the PC–SMC communication model. *Slave processing* delays the response by an additional random time. The response packet will be transmitted through *Net 2* into *SMC incoming buffer*. The response will be delayed until a token in the fusion place *Communications* is present. The token enables *Receiving*, so the processed message is put into *Received*. Eventually, the transition *From 2 to 1* will fire, what means that the second communication task has been executed successfully. Then, the same way, the next task for execution is chosen.

## 5 Implementation

The HTCP-nets model has been examined with CPN Tools [7]. Formal analysis and simulations have revealed that the concepts can be applied to manage numerous communication requests. This became a basis for development of software modules for the CPDev environment. The CPCon configurer (see Fig. 7) is used to define the horizontal communication between SMC and slave modules for a particular application. The CPSim tool collects program variables of a running SMC via Modbus and displays them on-line (*commissioning*).

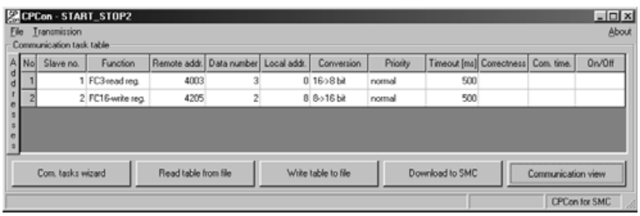



**Fig. 7.** Definition of communication tasks in CPCon tool

The SMC programmable controller is the first application of the virtual machine and the CPDev engineering package. Another application involves a PC equipped with an input/output board and a special software. The PC can be used as a *soft-controller*. A prototype solution which interfaces National Instruments NI-DAQ USB 6008 board to the CPDev environment has been also developed and tested [8].

## 6 Conclusions

Hierarchical approach allows for splitting a model into several subpages. This way, more and less detailed levels of the system description can be provided, simplifying development process. Low-level subpages can be analyzed independently from the high-level ones and easily modified. Additionally, multiple instances of a subpage can be created and used in the model repeatedly.

The HTCP-net models can be used not only to specify software modules, but also to estimate performance parameters. This can be important for small distributed control and measurement systems where single CPU handles both program execution and communications. In particular, the performance of vertical communication can be optimized. Register readouts in Modbus protocol can be either split into several transactions (one per each variable), or grouped into a single transaction involving continuous set of registers. The extra registers (not needed) have to be read anyway to make this set continuous and can be discarded later. Selection of the better way from the two depends on the number of not needed registers, baud rate, communication delay of a single message, controller cycle time, and average (or maximum) time of program execution (can be calculated). The CPSim tool of the CPDev environment uses such technique to minimize delays and optimize throughput.

Average and maximum response times can be calculated for typical cases. In this way possible bottlenecks can be predicted in the early stages. The simulations results have been compared with the real system and turned out sufficiently consistent in typical conditions.

## References


1. Rzońca, D., Sadolewski, J., Stec, A., Świder, Z., Trybus, B., Trybus, L.: Mini-DCS System Programming in IEC 61131-3 Structured Text. Journal of Automation, Mobile Robotics & Intelligent Systems 2(3), 48–54 (2008)
2. IEC 61131-3 standard: Programmable Controllers Part 3, Programming Languages. IEC (2003)
3. Jensen, K.: Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use. Springer, Heidelberg (1997)
4. Modicon MODBUS Protocol Reference Guide. MODICON, Inc., IAS, Massachusetts (1996), http://www.modbus.org/docs/PI_MBUS_300.pdf
5. Rzońca, D., Trybus, B.: Timed CPN model of SMC controller communication subsystem. In: Węgrzyn, S., Czachórski, T., Kwiecień, A. (eds.) Contemporary Aspects of Computer Networks, WKŁ, Warszawa, pp. 203–212 (2008)
6. Rzońca, D., Trybus, B.: Application of coloured Petri net for design of SMC controller communication subsystem. Studia Informatica 29(4B)(81), 15–26 (2008)
7. CPN Tools: Computer Tool for Coloured Petri Nets, http://www.daimi.au.dk/cpntools
8. Krok, K., Rzońca, D., Sadolewski, J.: Application of I/O boards in the CPDev engineering environment. Pomiary Automatyka Robotyka, 9–11 (February 2009) (in Polish)

# IEEE 802.11 Medium Access Mechanisms in Industrial Applications

Wojciech Domagała

Silesian University of Technology, Institute of Informatics,
Akademicka 16, 41-100 Gliwice, Poland
wojciech.domagala@polsl.pl

**Abstract.** The recent intensive development of networking technologies together with an increasing tendency to use broadband networks in industrial applications have caused the appearance of many new methods and protocols based on Ethernet. Rapidly expanding wireless branch of networking will certainly force even more problems on IT and Control professionals to solve. The specifics of wireless media, its vulnerabilities and new topologies are unfortunately an even match for it's undisputable features.

For that reason a basic analysis of medium access methods is made to verify IEEE 802.11, one of the most popular wireless computer communication standards, ability to be used in industrial or other time deterministic environments. Four methods of media access are taken into consideration in this paper: DCF, PCF, EDCF and HCF. Their basic features are confronted with industrial systems demands and each one is evaluated on possible fields of use.

## 1 Introduction

Wireless communication quickly conquers consecutive computer networks fields of appliance. Development favouring market demands concerning mobility, simplicity of installation, scalability and versatility caused unbelievable technological acceleration in this matter. Wireless networks not only allow to install computer devices where it was difficult to place wire network, they also significantly simplify the communication with movable and mobile devices. The last of the mentioned features – communication with mobile devices – makes wireless technologies future paaticularly promising. Wireless networks already allow relatively good mobility of clients in the covered area, mainly in the home or office solutions, but global tendencies show that industrial applications are also more often based or at least complemented by wireless technologies. The market is constantly introduced with new commercial solutions and approaches [1] and as the professionals experience is divided between them there are still no leading standards. If the wireless solutions career will be similar like this of industrial Ethernet it is almost certain that wireless technologies in industrial applications will be an area of intense interests, researches and changes in the coming years.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 272–279, 2009.

With such conclusion in mind this paper is an approach to a basic analysis of the most popular wireless local area network standard – IEEE 802.11 – in terms of use in industrial applications.

## 2 Medium Access Methods Analysis

As already stated – wireless communication in general offers many new opportunities to industrial computer systems designers and engineers. Taking into account the experience they gained while wired communication based on Ethernet developed [2] and gradually increased its share in industrial communication market, the IEEE 802.11 standard becomes a natural point of interest as a wireless extension of already existing and developing 802.3 based networks. This interest is particularly understandable as 802.11 was also designed to be used in local area networks and it is a part of 802 standard group so it is designed to be interoperable with other standards.

However, handling an uncontrolled medium is far more complex and challenging than the wired medium used by Ethernet . This complexity had a thorough impact on the Media Access Control part of the Data Link Layer (layer 2 of Open Systems Interconnection). The physical properties of medium, clearly different from Ethernet, and mobility – a feature naturally associated with wireless communication, forced a change in the protocol used. 802.11 utilizes CSMA/CA (carrier sense multiple access/collision avoidance) protocol which, although from the same family as CSMA/CD (collision detection), is more complex and restrictive [3].

These differences are caused by the inability to directly detect a collision in the wireless medium (like it is done in CSMA/CD in physical layer). To fulfil the idea of multi access and ensure the proper transmission of correct data a collision avoiding algorithm was designed. Its idea is divided in two parts:

- medium occupancy detection
- medium access method

The occupancy detection is the basic part, common for all media access methods. It is based on a dual mechanism which begins already in physical layer where it relies on a signal detection. It is supported by a software based solution and uses a virtual representation of signal called network allocation vector (later referred to as NAV).

Medium access methods are responsible for resolving the problem of several stations ready to use the medium at the same time. All of them are briefly described and analysed below.

### 2.1 Distributed Coordination Function

Distributed coordination function (DCF) is a basic and the most commonly used method of medium access control in 802.11 standard. It is implemented by all vendors which makes it a natural candidate for building cost-efficient

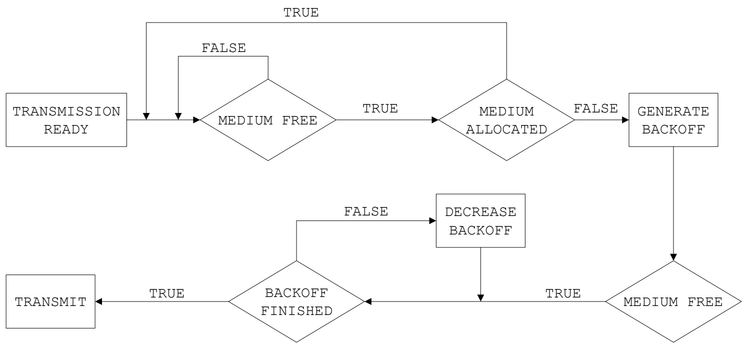



**Fig. 1.** Media access with use of Distribution Coordination Function

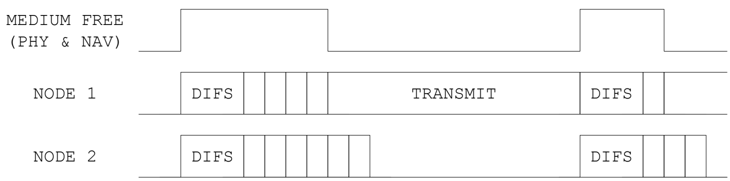



**Fig. 2.** Sample DCF scenario

(many devices to choose from) wireless communication system. DCFs another interesting feature is a complete distribution – network based on DCF does not need any special node acting as a coordinator or a similar central, key role (although it does not mean it cannot work in such mode).

Unfortunately this method has also a major drawback. The lack of coordinating node making status of all stations in the network equal results in a contention based method of accessing the medium. DCF uses a random backoff algorythm which chooses a value ranging from zero to a value called Contention Window ($CW$) which, in turn, can range from $CW_{\min}$ to $CW_{\max}$ parameters. The chosen number means a time (calculated in slot times[1]) which a station has to wait after detecting free medium before accessing it.

The method is not ideal even when it comes to the simplest case of two nodes in the network – it is probabilistic and because the floor of the range from within which the random number is chosen is always zero there is always a certain probability that the stations will start the transmission at the same time therefore leading to a collision. From the same reason the order in which stations will gain access to the medium is very difficult if not impossible to determine.

A sample scenario of medium access utilizing DCF is presented in Fig. 2. It presents the already mentioned simple two-node network. When the medium is reported free (no signal detected in physical layer and NAV counted down to

[1] A time value based on the radio frequency of the physical layer.

zero) both stations wait a constant time called DIFS (DCF inter-frame space) extended by the random backoff. Node 1 chooses lower backoff and gains access to the medium. After the complete transfer (data and acknowledgement – acknowledge mechanism is not covered in this paper) the medium is again reported free and again Node 1 gains control over the medium choosing lower backoff. Node 2 can try to gain access when Node 1 finishes its second transmission and it should eventually succeed but a significant increase of waiting time is possible.

### 2.2 Point Coordination Function

Point coordination function (PCF) is an additional method of medium access control described in IEEE 802.11 standard. It is not as popular as DCF because its implementation is not mandatory according to the standard compliance demands and is often omitted by vendors. In contrary to the DCF mechanism, PCF does not allow network nodes to access the medium freely. It provides a mechanism of contention free data transmission in a network where one of the nodes acts as a Point Coordinator (it is an Access Point usually) controlling other PCF-supporting nodes. The contention free period begins when the coordinating node announces it using a special control frame. All the nodes in the network, after receiving such an information switch themselves in PCF mode and their access to the wireless medium is from then centrally controlled by the coordinator. The coordinator can send data to the nodes and poll the nodes to retrieve data from them as well. The PCF finishes when the time announced in the starting frame runs up or when Point Coordinator finishes it with another control frame (whichever comes first).

The PCF communication scenario clearly resembles the popular Master-Slave [4], [5] paradigm well known among control engineers and widely used in industrial networks and applications. The medium access strictly controlled by Point Coordinator ensures that every station will access the medium and reduces the risk of a single station dominating the medium only due to propitious backoff

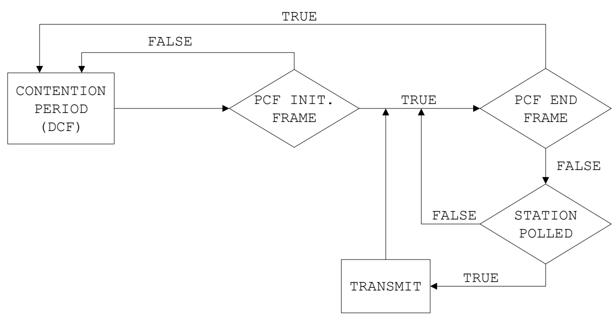



**Fig. 3.** Point Coordination Function

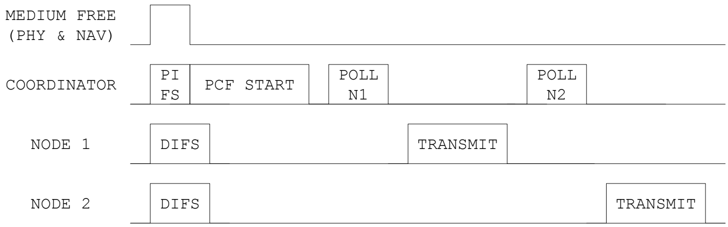



**Fig. 4.** Part of a sample PCF scenario

generation. However PCF is not deprived of disadvantages. One of them, being a feature responsible for the main advantage at the same time, is centralized medium access control – single coordinating point makes the network more susceptible to device damage resulting in a complete halting of communication. The PCF mechanism also adversely influences the maximum network throughput generating additional control traffic.

Figure 4 pictures an excerpt from a sample beginning of PCF scenario. Every node activity has to be preceded by a proper coordinator control frame. The beginning of the contention free period is particularly interesting as it can occur in any moment while the network is working in DCF mode – coordinating node does not wait DIFS time like a DCF node but PIFS (priority inter-frame space) which is shorter. This mechanism allows the coordinator to start PCF mode at any time with absolute certainty (after the current transmission is over and the medium is reported free). Figure 4 does not include the end of PCF mode but it is quite simple – when coordinator communicates with all nodes it sends a PCF ending frame after which DCF mechanism is resumed.

### 2.3 Quality of Service Extension

Beside DCF and PCF, being two basic and quite simple methods of accessing medium, IEEE 802.11 also includes a Quality of Service (QoS) extension. The main idea standing behind it was an improvement of MAC layer to enable delay-sensitive services to be run in the network. The QoS technology itself is already well known from the wire communication but similarly to other medium access mechanisms it faces different problems in its wireless version and as such is treated as a quite new. Quality of Service proposed for 802.11 includes two following methods:

– HCF (Hybrid Coordination Function) with contention (also known as EDCF – Enhanced DCF)
– HCF with polling

Although QoS extension was designed with the main purpose of voice and video applications support its features may also be helpful during design and deployment of an industrial wireless network.

**Enhanced Distributed Coordination Function.** The idea of this DCF extension is based on differencing not only network node priority (which can be done approximately by parameterizing $CW_{\min}$ and $CW_{\max}$ parameters) but also the data sent. It divided data to eight categories called traffic classes and assumed that communications system utilizing QoS will be complemented by a data classification mechanism (not described in standard).

Properly classified data are placed in different queues where they await for the transmission. However prioritizing data frames in a network of client devices not communicating with each other directly is not a trivial matter. To solve it EDCF introduced several changes and improvements:

- transmission opportunity (called TXOP)
- arbitration inter-frame space (called AIFS)
- different $CW_{\min}$ and $CW_{\max}$ sets for categories

TXOP is a time when the node can transmit frames. It is long enough that the node can transmit more than one frame without participating in usual DCF contention after each frame.

The AIFS is a time span which a station waits after detecting free medium, it is very similar to the DIFS and PIFS times but is adjustable and the shorter set the higher priority of transmission and better chance of accessing medium first.

Changing $CW_{\min}$ and $CW_{\max}$ influences the chance of accessing the medium during the regular DCF scenario – the $CW$ value is chosen from different ranges, depending on data class.

**Hybrid Coordination Function (with Polling).** The idea of HCF is based on a principle similar to already described PCF, it accesses the medium with PIFS space but uses TXOP mechanism as well allowing the network node to send more than one frame after polling.

Quality of Service technology also provides a mechanism called admission control which is responsible for controlling the network throughput, current free resources (called available budget) and distributing information across the network (in EDCF) or using it when generating polling schedule in HCF method. It allows to control the data flow in the period of transmission requests reaching the maximum network throughput.

Although the mechanism is interesting the industrial applications are usually well described at the stage of design in terms of amount and frequency of data transmission so admission control is of less significance for them and quite complex therefore not covered in this paper.

## 3 Summary

These several briefly described above methods unquestionably requires accurate analysis, tests and researches to precisely define which fields of appliance they would fit best. However even after this short study of each some tendencies and probable research directions can be observed.

An obvious candidate for applying in industrial applications is the PCF method which way of working resembles strictly determined network exchanges scenarios. Clearly all other mechanism based on the PCF principle are equally interesting. The one described in Quality of Service extension enables a variety of parameters to be changed and influence the transmission of different types of data from different types of nodes.

Beside the PCF alone also the mixing of the method with one of contention based to create a time window for asynchronous data exchange for example. Such hybrid solutions, allowing the creation of communication systems meeting high demands of industrial computer systems, slowly appear on the device market and are a field of intense research as the IEEE 802.11 does not include and standardize all these parts (for example already mentioned data classification mechanism). Thanks to that the standard is flexible and adaptable to control engineers demands.

As to the contention based mechanisms – using them alone is possible rather only in small networks without significant traffic load and not very strict technological demands. However it would still be better to use EDCF method as it is much more capable in terms of parameterization of transmission than DFC.

Contention mechanisms could also find themselves applied for mobile HMI (Human Machine Interface) devices where the speed of the network greatly exceeds human reaction and very fast communication is not the major issue.

Despite not very insightful analysis IEEE 802.11 proved to be a promising technology in terms of use in industrial applications. Properly chosen medium access methods together with proper parameterizing paired with high bandwidth can result in a versatile, efficient computer network providing us with more features then wired network.

It is also important to mention the economical problems spreading around most of the globe. Their scale will certainly influence the number of investments in the nearby future and probably the pace of new technologies development. Rapidly decreasing income may cause both reduction of control systems market and reduction of financial support of research centres, which in turn can cause the pace of specialized solutions development of to slow down. From these we can conclude that an urgent necessity to seek new solutions emerges. The mass consumer market, where strong competition provides respectively low devices cost, is probably being able to encourage investors to keep improving and expanding their control systems, even during the period of the economy slowing down. This is another reason to analyse, adopt and release the potential of 'consumer technologies' in industrial applications.

## References

1. Willig, A., Matheus, K., Wolisz, A.: Wireless Technology in Industrial Networks. IEEE, Los Alamitos (2005)
2. Breyer, R., Rileyi, S.: Switched, Fast and Gigabit Ethernet. Macmillan Technical Publishing, Basingstoke (1999)

3. Roshnan, P., Leary, J.: 802.11 Wireless LAN Fundamentals. Cisco Press (2003)
4. Kwiecień, A.: Analiza przepływu informacji w komputerowych sieciach przemysłowych. Wydawnictwo Pracowni Komputerowej Jacka Skalmierskiego (2003)
5. Grzywak, A., Kwiecień, A., Gaj, P., et al.: Laboratorium Sieci Komputerowych. Wydawnictwo Politechniki Śląskiej (1999)
6. IEEE Std 802.11-2007, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE, Los Alamitos (2007)

# The General Concept of a Distributed Computer System Designed for Monitoring Rock Movements

Piotr Gaj and Błażej Kwiecień

Silesian University of Technology, Institute of Informatics,
ul. Akademicka 2A, 44-100 Gliwice, Poland
piotr.gaj@polsl.pl
http://www.polsl.pl

**Abstract.** In this article the authors consider a construction of a computer distributed system designed for monitoring geological phenomena such as big rock falls[1]. In order to create universal IT solutions for this kind of threats, authors designed a general concept of a system without using an off-the-shelf expensive equipment. This is the first step to building a prototype of such a monitoring system as well as conducting additional research related to measurement of dislocation and low power transmission.

## 1 Introduction

Using computer technologies is obvious, with regard to any kind of human activity. Therefore, computer systems are useful just like we want to assist various measurements connected with geological phenomena and for collecting data and its analyses in order to create a forecast of natural events. A very important type of these phenomena are soil and rock instantaneous falls, because of their danger. The paper refers to monitoring of this type of geological threats with IT techniques. To forecast the behaviour and predict the hazardous and final slide of a rock, some datasets with regular measurement values of rock dislocation as well as prediction algorithms are needed. In order to collect and process data, an automatic unaided and cheap system should be created. Similar systems exist but scopes and ways of using them are different than presented in this paper and their cost is very high. Thus, authors created a concept of such a system from scratch.

In this article we are going to present the idea of a distributed wireless computer system designed for cyclical monitoring of the rock dislocation. A periodical collection and storage of this kind of data can be useful for statistical analysis and could be helpful for creation of predictions. The main components of such a system are locally spread measurement stations and a central database server. Local systems should act as field objects whereas a database should be a

[1] Topic connected with "Geology & Information Technology" seminar project organized by The Polish-Norwegian Research Fund, Silesian University of Technology and Sogn og Fjordane University College.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 280–291, 2009.

data repository for all of local systems and should contain an interface to other, external systems.

In case of a rock fall the most related topics include: the methods of precise dislocation measurement, local robust wireless communication, long distance communication as well as local and central data storage. Additionally, the power supply and mechanical construction issues of every part of a local system are very important because the system has to act as unaided automaton for at least one year in very diverse and extreme weather conditions. The scope of this paper covers the whole structure of the local measurement system (LMS) and the communication issues. Additionally, a set of possible measurement methods of movement is presented, especially those based on the GPS.

## 2 Local System Architecture

In the single local field area there are typically a few measurement points spread along a given region with about 100 metres in diameter. The main task is to measure the distance changes between any two given points. It is assumed that the local region contains a stable and unstable areas. The locations of measurement points have to be fixed because of measuring of the dislocation between them with precision of at least 1 mm. Therefore, it is possible to fix a part of them on the stable rock and the others on the unstable parts of the local area. The points located on the stable rock are the reference points, to which all measurements refer. In expected scale one reference point should be enough. In the LMS three functional tasks can be distinguished. The first one is to measure a relative movement in the given point. The second one is to collect data from all measurement points and transmit it to the central database located in the distant place. The third one is to process the local data and, depending on the achieved results, to control the local actuators. This is illustrated in Fig. 1.

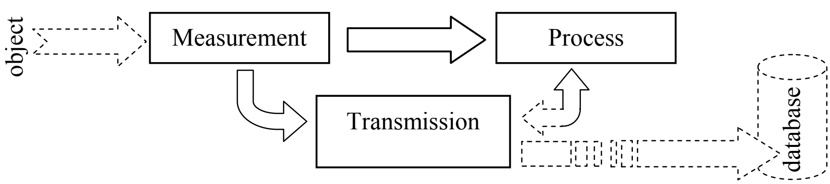



**Fig. 1.** Basic tasks of the LMS and its dependencies

### 2.1 Distribution of Functions

The local system should contain a set of nodes with communication capabilities. Each of the nodes has a processing unit with an interface to measurement appliances and is able to use it, get a local measure and put it into its memory. One single system node neither can process the whole system information nor send

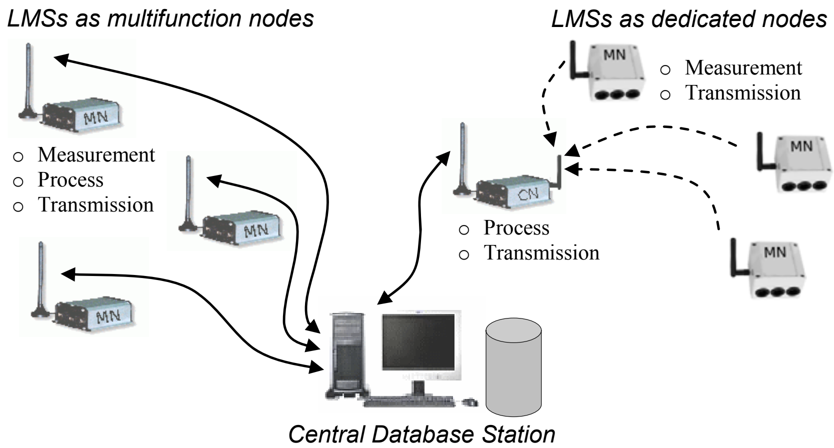



**Fig. 2.** Proposed solution for distribution of system functionality

it to the distant database because it does not have the whole set but its own one. To resolve this problem there are two ways of designing the system node. The first one is to create universal node, with the ability to execute all three functional tasks. The second one is to create a specialized node to which the execution of the functional tasks is delegated. To realize the first solution, the local data from each node should be transmitted to the others in order to gather complete information about the state of the local system. Thanks to this concept an ability to use redundancy and dynamic reconfiguration of the local system exists. In addition, if the long distance communication capability is added to the node functionality, each node has a full functionality of a local system and is able to relieve any other node, in case when it is broken. However, in the LMS considered in this work there is no special reason to add advanced individual features to each node and it can be assumed that only one extended station is needed. Therefore, broadcast transmission as well as embedding complex processing and a high power radio functionality in each system node is unnecessary. Thus, the better solution is to create nodes according to the second idea mentioned. In order to put it into effect, a special devices called Measurement Node (MN) and Central Node (CN) can be introduced. The main components of MN are measurement unit and local communication unit. The main parts of CN are powerful processing unit, local communication unit, and distant communication unit. Nevertheless, the reliability of functionality maintaining by the local system has to be taken into consideration. Functionality reduction is not a good solution because it discards the possibility of using the previously mentioned redundancy. The multiplication of functionality in the system nodes gives an option to set up the system as well as to reconfigure it in case of malfunction of any node. But finally, there are two important necessities to reduce functionality of a given node because of power consumption and its financial cost. Mentioned issues are presented in Fig. 2.

### 2.2 The Communication Channel

There are several solutions to use with a short and long distance communication including cables. Because of the cost of cable infrastructure and its mechanical and lack of environmental resistance, the cable connection idea has been rejected and it is assumed that only a wireless channel based on the radio transmission is further considered. Additionally, there is a assumption of using public free bands to avoid potential problems with acquiring appropriate licences. It is possible to use a band named ISM, which contains sub-1 GHz and 2.4 GHz frequencies as well as GSM network based on 900 MHz, 1.8 GHz, 2.1 GHz and others. The range and accessibility of sub-1 GHz and GSM bands differs according to the local regulations. But generally, 2.4 GHz is a public band everywhere and GSM network is accessible almost everywhere where people live. Generally, in the LMS there is one simple communication task. The local set of variables from MN should be delivered to the repository in the central database (CDB). Thus, two types of data passing can be discussed. The first one is to use long distance transmission from each node to the distant station, and the second one is to use short distance communication between each of the local nodes and the main station. This station is also local but it is a specialized node with a capability of using long distance transmission based on techniques mentioned previously. The second one seems to be better because of the conclusion from point 2.1. Distant communication from each of nodes to the database server is not effective from the power consumption point of view. Let us assume a typical system which contains four measurement nodes. The radio in typical radio modem uses the transmission power of about 5 mW – 1 W with the operational current of the whole device being approximately 40 mA – 611 mA. The radio, in GSM case, uses the transmission power of maximum 1–2 W with the operating power of modem about 175 mA – 1.7 A. Local transmission needs power of about 1 mW and operating power from 5 mA up to 40 mA in Bluetooth, ZigBee or general ISM case [13,12]. The sleep mode power consumption is about 16 mA for radio modems, about 5 mA for GSM modem and about 0.5 $\mu$A for ISM devices. The time of transmission of a few local values of variables depends on transmission type and data size. Let us assume eight variables of 16-bit which should be enough to code a few measurements. Generally, in any case before transmission of useful data there is a necessity to establish a logical channel. This is time-consuming in GSM case and takes about 4–6 seconds. Time of data transmission of 32 bytes is below 3 ms and it is not important with relation to time of channel establishment when transmission speed is in range from 1 Mb/s down even to 20 kb/s. Generally, a period of 6 seconds of radio activity per each data subset transmitted by the node can be assumed. ISM solution requires roughly from a few micro seconds to a few milliseconds for transmission activation. Thanks to the shorter time of radio activity and less power consumption in sleep mode the total power consumption is lower in short distance transmission case. Graphical depiction of the power consumption by various device types is presented in Fig. 3. The conclusion is obvious. MNs need less power for local transmission than for the distant transmission. Thus, this is the main reason for creation of a special

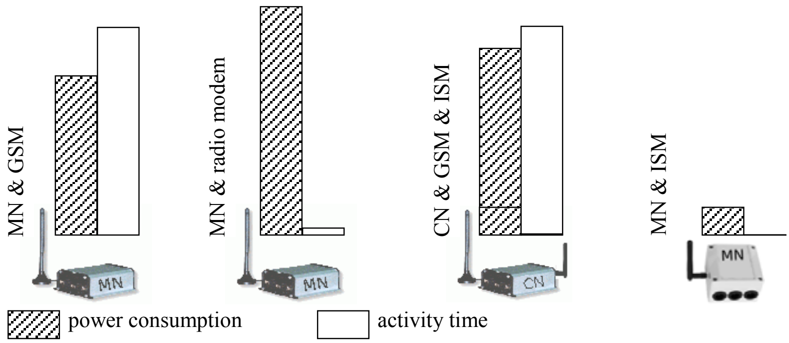



**Fig. 3.** Estimated comparison of typical power consumption and radio activity time during transmission

CN dedicated to local data collection and which later retransmits it to the CDB station. Only this node has to have a more stronger power source designed for communication with CDB.

If the multifunction node idea is rejected, there are two additional types of transmission: local and distant. In the local channel case there can be distinguished a few popular, low power and cheap standard techniques included in IEEE 802.15 WPAN, IEEE 802.11 WLAN, open source projects and commercial solutions like: Bluetooth, ZigBee, CC1000, Z-Wave, Insteon, EnOcean, OneNet, 6lowpan, Hiperlan, Dect and others [11,12]. To establish the distant channel there are two technical possibilities: either a radio line with radio modem or a cellular network with GSM modem.

**The Local Channels.** In order to establish a local channel, the ZigBee personal network is chosen from above. The main advantages for that selection is low power consumption and low cost of hardware components, good knowledge, standardization and hardware element base and a relatively big radio radius up to 100 m. Additionally, this standard gives a possibility of using multihop routing with beacon servicing and also the general network architecture fitting to the system architecture presented above. There are three kinds of devices in ZigBee:

- Coordinator (ZC) – acting as a network supervisor and a security centre. Only one may exist in the given network. Any other network subscriber can attach to this device and send data. In LMS, the functionality of coordinator agrees with the CN. Thanks to that the node is able to act as a repository for measurements, network configuration and security keys.
- Router (ZR) – acting as a data retransmitter. In cases when the distance between nodes has to be about or longer than maximum radio range, router can read the data from one node and transmit it to the other. For the LMS the topology is fixed and it does not need to use any dynamical routing algorithms known from sensor networks. Nevertheless, the router should pass data from the given MN to the other given MN according to fixed routing

algorithm if the location of measuring points is out of range. Router can act either as a special type of system node dedicated only to transmission purposes or as an MN.

- End Device (ZED) – acts as data transmitter. It cannot receive useful data from others or relay data to others. It can pass data only to a parent node which can be a router or a coordinator. For LMS this device belongs to MN. Most of the running time this device can be asleep, because of periodical character of data transmission.

In order to put the CN to sleep, the beacon servicing should be used. Beacon intervals for low speed transmission of 20 kb/s may be in range between $48 - 48 \cdot 2^{14}$ ms. Therefore, all communication devices in nodes can be put to sleep mode and wake up only with a given period of time in order to exchange data. Let us assume exchange scenario related to the previous LMS example. The measurement cycle depends on movement level per time unit. If a value is below assumed tier, the cycle is of 60 s, and if above, it is of 15 s. Of course, there is an option to consider more than one tier and build a more complex dynamic scenario. So in example of LMS, a packet of 32 bytes of useful data is transmitted every minute from each node. It equals to 525600 times per year, equivalent of 16 Mbytes. To calculate the real total number of transmitted data, the protocol control data has to be added from ZigBee and 802.15.4 stack. That control data is necessary to realize ZigBee services such as starting network, joining nodes, reception and confirmations, routing etc. Thus, in order to calculate precisely the time of network activity, the time analysis of ZigBee protocol is needed and this is out of scope of this paper. The topology of LMS example, based on ZigBee devices, is presented in Fig. 4.

There is another way of establishing an exchange scenario. The measurements can be gathered together in MN memory and sent with a longer cycle. The disadvantage of this solution is that CDB does not receive data with a measurement cycle but with a network cycle. But the advantage is great because the communication device can be longer in a sleep mode. In addition, for relative big packages it is possible to use data compression and reduce a packet size by several percent and thanks to that also reduce radio activity time and power consumption.

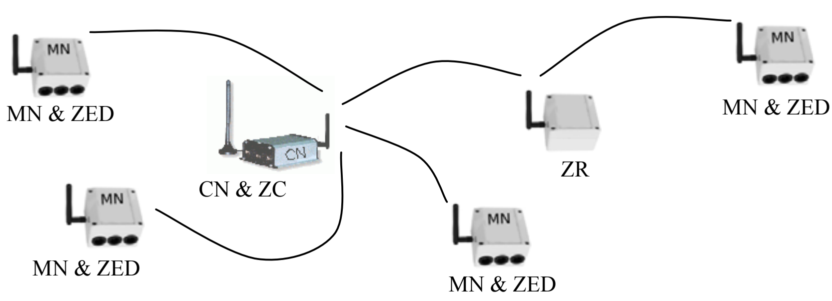



**Fig. 4.** Example of LMS with ZigBee local communication

Mechanism of data gathering and compression can be turned on and off depending on criteria from the idea of dynamic cycle presented above. In general, ZigBee is one of the options. Other solutions can also be used including creation of universal CN, either within the given subset of network solutions or as a kind of ambient network supervisor station. This universal way of connections allows for utilization of a variety of MN from various producers.

**The Distant Channel.** As it was mentioned previously, there are two ways of long distance transmission: dedicated radio line, or GSM network. In the first case there is a necessity to install radio modems in a field with unobstructed field of view. This is the technical requirement. It can be difficult when the region is arborescent, hilly, or really vast. In this case the repeaters should be installed either for passing signal out of the hill or to reinforce it. It could be economically unprofitable because the radio modem devices are relatively expensive. However, there is one advantage of this solution. The transmission is free of charge so the maintenance cost of the system is low. In the second case one does not need to use an expensive radio modem, instead of this a cheap GSM modem could be installed [3]. In order to send data through GSM network, GPRS can be used as a good service to integrate cellular network subscriber with IP-based network with CBD. However, there is extra charge for sending a data packet through GPRS channel, apart from a network provider subscription fee. The problem of data cycle transmission establishment is similar to the local transmission case. There is a possibility of gathering data and postponed sending as well as instantaneous sending in case of detection of outpassing the tier. Answering the question which solution is cheaper is difficult because of constant fluctuations of devices and services costs. Nevertheless, the simple balance is shown below. Let us calculate roughly the cost of the initiation of the example system and its annual maintenance cost. The cost of a simple radio modem with a repeater functionality is about 700 Euro, the cost of a GSM modem is about 20 Euro. Let us assume a plan for data transfer with fee on level of 30 Euro and no connections fees[2]. Additionally, let us simulate the previous example of LMS with a simple case of two radio modems. The cost of devices is about 1400 Euro and 20 Euro accordingly. For the radio modem the transmission cost is either zero or depends on the cost of obtaining the licence. For the GSM modem the operational cost is about 360 Euro per year. However, the cost of the radio modem can be increased significantly if the distance between modems has to be greater than the maximum reach distance coming from transmitter power and receiver sensitivity. The solution with GSM seems to be more attractive because of the cost of devices. Maintenance cost depends on local providers' plans. The schema of distant transmission based on GSM is presented in Fig. 5.

To integrate LMS with CDB, the Internet is the best solution. The ideal case is when all stations have their own static and public IP address. This is not always possible because of shrinking unoccupied address space of IPv4. Using this version of IP is necessary in respect of common devices compatibility. Therefore,

[2] Sample data based on typical market offer on February 2009.

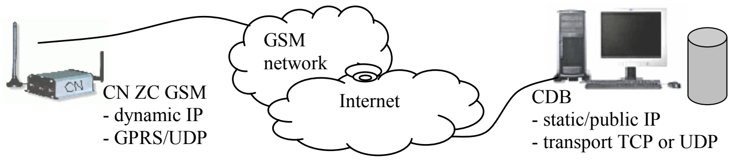



**Fig. 5.** Long distance connection via GPS and Internet

in one way communication at least the CDB station should have a static and public IP address or address routed in a given system subnet. Then CNs are able to send data directly to the given socket. If transmission is bidirectional and CNs have dynamic addresses, a special routing mechanism is needed [6,7]. It can be located in CDB. Using the UDP protocol in transport layer reduces the number of bytes transmitted through GPRS channel. It is also possible to use a reliable TCP protocol instead of a simple UDP. Although the TCP protocol is safer because of establishing the logical connection between stations, its transportation overhead is bigger. It may have a significance if GPRS transfer is paid per packet.

## 3 Measurements

There are several potential methods of measuring a dislocation. The first group contains classical sensors based on optical, mechanical and acoustic detectors. For instance it can be: shock loggers, geophones, extensometers, laser devices etc. They have appreciable virtues as:

- low cost of devices and their installation,
- very high accuracy of measures,
- low power consumption.

Unfortunately there are disadvantages as:

- the measurement can be disturbed by any object in the vicinity of the sensor,
- the measurement methods are not weather proof.

Those disadvantages make it impossible to use because of environmental requirements. The second group contains sensors based on satellite positioning systems as GPS, together with a standard receiver based on raw satellite signals without any correction. Virtues of this solution:

- low cost of devices and their installation,
- measurements independent from weather (except extreme conditions).

There is one important disadvantage of this: low accuracy of measurement (up to several meters). On the contrary, the GPS with a differential and/or static method has virtues of standard GPS and very high accuracy of measurements. However, there are also a few disadvantages as:

- high cost of devices and their installation,
- long time of measurement process,
- need to synchronize clocks of MN and CN,
- highest power consumption among other concepts.

Because of unavoidably of weather fluctuations, the best solution seems to be the GPS system. The GPS system is designed in the way that allows to observe at least 4 satellites in every point on the Earth ground. It gives a possibility of measuring a distance from a given point to the four satellites which location in the space is known. Two levels of navigation and positioning are offered by the Global Positioning System: The Standard Positioning Service (SPS) and the Precise Positioning Service (PPS). The Standard Positioning Service offers a base-line accuracy that is much lower than the PPS, but is available to all users with even the most inexpensive receivers. The Precise Positioning Service is a highly accurate positioning. Using only a single C/A-code receiver we can take accuracy around 10 to 30 meters. To get better accuracy we can use differential GPS positioning (DGPS). The underlying premise of differential GPS (DGPS) is that any two receivers that are relatively close together will experience similar atmospheric errors. DGPS requires that a GPS receiver be set up on a precisely known location. This GPS receiver is the base or reference station. The base station receiver calculates its position based on satellite signals and compares this location to the known location. The difference is applied to the GPS data recorded by the second GPS receiver, which is known as the roving receiver. From the difference between the known position and the GPS-derived position differential can be calculated, which can then be applied to the rover's position data. Another very important consideration is the distance between the base and rover. This is because if the rover is a substantial distance from the base, it's possible that the two receivers might be observing one or more different satellites. For code differential positioning, base line distances should be kept below 300 kilometres [10]. The higher precision is riched by using static positioning technique. Static positioning implies that both receivers are stationary during the entire period of data collection. The length of the observation period is dependent on parameters such as the number of observed satellites, the distance between two receivers, receiver type and accuracy requirement [1]. Static GPS using carrier phase observations and simultaneously occupying sites for an hour or more is sometimes referred to as conventional static GPS, since it is the original technique used for GPS surveying. In this technique receivers track the same satellites simultaneously for at least one hour. One of the main reasons for occupying sites for over an hour (sometimes several hours) is to exploit the change in geometry as satellites track paths across the sky. It is this change in geometry which assists in ambiguity resolution and helps to improve the strength of solution. The range of accuracy using conventional static GPS varies depending on the observing and processing procedures followed, the baseline lengths measured and the receivers used, among other variables [9]. In static relative positioning the two receivers must remain stationary throughout the observations while vector between points A and B is being computing [14]. Differential corrections are used to compute the carrier phases. First-approximation measurement for the distance between a satellite and a navigation satellite receiver is pseudorange. A few methods exist to compute pseudorange but the most precise is phase method. How to compute pseudorange using phase is shown below:

$$d = N \cdot \lambda + \lambda \cdot \varphi \tag{1}$$

where:

$d$ – pseudorange,
$N$ – whole number of waves length which are include in range between satellite and receiver,
$\lambda$ – wave length,
$\varphi$ – incoming signal phase.

There are many GPS devices designed for precise positioning for geodesy purposes [15]. One of the most interesting suitable for LMS is GPS Module – ProMark2. It uses static survey performance (rms) with following accuracy specifications: horizontal 0.005 m + 1 ppm, vertical 0.010 m + 2 ppm, azimuth < 1 arc second. Nominal observation time is in range from 20 to 60 minutes depending on distance between ProMark2 receivers and other environmental factors. These parameters and high cost of about $8000 discards such a devices as a good measure. There is necessity to do some research with classical GPS receivers, DGPLS and/or static method, and local reference station.

## 4 Power Supply

Considering hard accessibility of field locations as well as limitations and snags related to the cable installation, in described LMS car batteries are used. Additionally, the possibility of charging with solar energy is considered if using solar panels is possible in given environmental conditions. Installation and maintenance cost has also an influence on the battery solution selection. This cost is significantly lower in comparison with energizing the system by an external and permanent power supply source. Solar panels are built from fotovoltaic cells and work to charge batteries or to energize low power devices directly. The amount of produced energy depends on solar radiation intensity, so using of solar panel depends on location of a field. It is especially important when LMS is located in the area with long nights. According to the mentioned example, let us estimate the life time of unaided MN power supply with pessimistic assumption:

- for communication unit:
  - the integrated element based on CC2430 or similar [13] could be used,
  - activity time, during preparing and making one packet transmission, of 10 ms,
  - 35 mA of operating and not significant 0.5 $\mu$A for sleep power consumption,
- for processing unit:

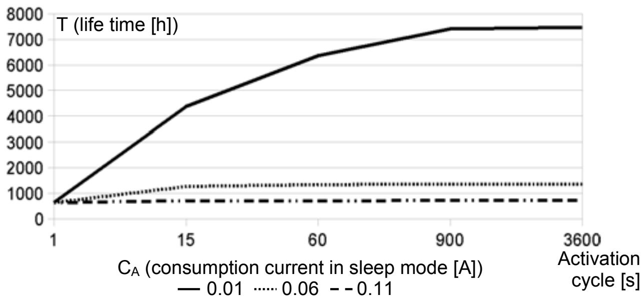



**Fig. 6.** Working time of MN according to cycle and sleep power consumption

  - the unit based on AVR processor [5] can be used,
  - processing time, during measurement and transmission, of 1 s,
  - 50 mA for operating mode,
- for measurement unit:
  - the embedded GPS module can be used i.e. EM408, FGPMMOPA2 [2,4],
  - measurement time of 1 s,
  - power consumption of 67 mA,

and from 10 to 110 mA of total asleep power consumption and cycle of activation time from 1 s up to 1 h. With a battery of about $Q = 75$ Ah capacity, the estimated life time of the assumed device is presented in Fig. 6. To prepare results the average current consumption (CA) was calculated according to unit type and time of activation in a given supplying mode, either operating or sleep. The life time is calculated as $T = Q/CA$.

## 5 Conclusions

There is a necessity to design and create a computer system able to monitor the geological phenomena of rock falls. Nowadays, there are technical possibilities to make up such a topology, functionality and dependencies of hardware and software to build up an unaided monitoring system with high reliability. However, there is a problem to obtain a low cost and high precise of measurements. There are three problems related to this. The first one is a method connected with a highly precise dislocation measurement. It is essential to do research related to measurement method based either on the GPS or other positioning system (i.e. Glonass, Galileo) or to try to use a few systems together (i.e. based on G3 technology from Topcon). The relative, differential GPS method combined with the static method seems to be good enough but it also requires some research on small areas. The potential problems are the time synchronization between nodes and the duration of active state. The second one is power supply. A system has to be optimized in power consumption. Especially in power consumption in sleep mode, what one can observe in Fig. 6. The third one is weather conditions.

All electronic elements have to be designed to work with a wide temperature range including far below zero degrees Celsius. However, there are elements and embedded modules designed for extreme environmental conditions.

## References


1. Satirapod, C.: Improving the GPS Data Processing Algorithm for Precise Static Relative Positioning. Univ. of New South Wales (2002) ISBN 9780733419010
2. EM408 GPS receiver Engine Board. USGlobalSat, Inc. USA (2005)
3. Enfora Enabler II Specification. Enfora (November 2008)
4. FGPMMOPA2 Datasheet. GlobalTop Tech Inc. Taiwan (2007)
5. Gaj, P., Holodok, A., Bigewski, Z., Kwiecien, A.: Sterownik telemetryczny. Techniczne i spoleczne problemy zastosowania Internetu, WKL (2005) ISBN 8320615909
6. Gaj, P., Holodok, A., Bigewski, Z., Kwiecien, A.: Zdalny monitoring z wykorzystaniem sieci GSM. Techniczne i spoleczne problemy zastosowania Internetu, WKL (2005) ISBN 8320615909
7. Gaj, P., Kwiecien, A., Bigewski, Z., Holodok, A.: Adresacja IP w systemach rozproszonych wykorzystujacych komunikacje GPRS. In: Nowe technologie sieci komputerowych, WKL (2006) ISBN 8320616107
8. Gaj, P.: Zdalne monitorowanie obiektow z wykorzystaniem sieci komorkowych. In: Napedy i Sterownie, Raciborz, vol. 2(94), pp. 50–52 (2007) ISNN 15077764
9. GPS positioning guide, Canada (1995) ISBN 0600159171
10. French, G.T.: Understanding the GPS – An Introduction to the Global Positioning System, Baker GeoResearch (1997) ISBN 1566902258
11. Holger, K., Willig, A.: Protocols and Architectures for Wireless Sensor Networks. Wiley, Chichester (2005)
12. Kuorilehto, M., Kohvakka, M., Suhonen, J., Hamalainen, P.: Ultra-Low Energy Wireless Sensor Networks in Practice: Theory, Realization and Deployment. Wiley, Chichester (2007)
13. Low Power RF. Texas Instruments (2008)
14. Dikshit, O.: Modern Surveying Techniques (Web Course): Global Positioning System. Indian Institute of Technology Kanpur
15. Pudlo, M.: Niebo dla geodezji. In: Geodeta – Magazyn Geoinformacyjny 11(126) (November 2005)
16. ZigBee Specification. Document 053474r11. ZigBee Standard Organization (2007)

# Remote Monitoring of Geological Activity of Inclined Regions – The Concept

Jarosław Flak, Piotr Gaj, Krzysztof Tokarz,
Stanisław Wideł, and Adam Ziębiński

Silesian University of Technology, Institute of Informatics,
ul. Akademicka 2A, 44-100 Gliwice, Poland
{jaroslaw.flak,piotr.gaj,krzysztof.tokarz}@polsl.pl
{stanislaw.widel,adam.ziebinski}@polsl.pl
http://www.polsl.pl

**Abstract.** In the article authors consider the possibility of using current IT technologies to create a computer system dedicated to monitoring geological phenomena. Nowadays, there are a few technical solutions dedicated to monitoring various kinds of events. Some of them are standalone devices, other a little bit more complex systems. After seminar, realized together with geologists within the GIT project[1], the conclusion was born that there are no full-scale, comprehensive systems which can be able to perform a measurement acquisition at an field location, transmit information to a database, and process achieved data online and offline in an unaided way. Therefore, in the article the concept of such a system is presented. The system is dedicated to monitoring rock slides and similar geological activities. There are some techniques involved, which allows to reduce the processing and transmission time of data as well as power consumption and maintenance cost.

## 1 Introduction

Nowadays, information technology gives us an opportunity to gather a huge set of apparently useless information as cyclically collected measurements of various physical parameters. This information can be processed online as well, what is even most valuable, as offline. Thanks to that, we are able to achieve some regularities and correlations between values in time and next create predictions of a real behavior of the observed object. This could be useful in observations of geological activity of soil and rocks i.e. slope movement, rock avalanches and other dangerous phenomena occurrences on inclined regions. One of the most precipitous and perilous geological phenomena are rock slides. An occurrence of this phenomenon depends on many factors: rainfalls, pressures of water inside a

[1] Geology and Information Technology – seminar project organized by The Polish-Norwegian Research Fund, Silesian University of technology and Sogn og Fjordane University College.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 292–301, 2009.

rock and soil, earthquakes and micro-tremors and many other. These phenomena should be and thanks to nowadays IT techniques can be monitored.

In this article authors are going to present a concept of a distributed measurement computer system based on the sensor networks idea, designed for collecting various data from a given local area, its local preprocessing, local controls and sending gathered data to the central database system. Gathering information and processing it is executed in a cyclic manner with a parametrisable period. The main part of such a system are local measuring nodes – sensors (LMN) and a local process and control node – field computer (LPCN). Additionally, the central database server (CDB) has to exist but it can be located in a distant place. Communication with this server is established via GPRS channel because of low cost of devices, installation and maintenance [4,5]. It is assumed the measurements are performed by proper sensors and signal from it comes to system nodes as analogue voltage or current signal. It means that the system nodes are equipped with standard inputs with AD converters. Thanks to that, nodes are independent from type of measurement and sensor and could act as sensors for various purposes. The nodes are powered by batteries because of lack of power supply sources in the system's working area.

Any measurements should be executed in time function in the way that one can detect changes of factors. Generally, observation of geological phenomena needs ages, years, months. However, in case of rapid and fierce phenomena this is necessary to perform monitoring and data processing in real time with time limitations of days, hours, minutes, seconds. Such an approach gives a possibility of informing about upcoming outpass of assumed tier values for each measurement or about similarity of set of measurements to alarming pattern. As a result, it gives an opportunity to produce a warning or an alert of possibility of an avalanche occurrence.

## 2 Local System

In the single local field area there are typically several measurement points spread along the given inclined region, i.e. a slope. The size of measured area is about several hundred square meters. The locations of measurement points are fixed and/or variable. The main task is to measure and analyze the physical parameters as pressure, humidity, rainfall, movement etc. There are four functional tasks. The first one is to measure something in the given point. The second one is to transmit data to LPCN by each LMN and collect it in LPCN memory. The third one is to process the local data in run-time mode and, depending on achieved results, to control the local actuators. The fourth one is to transmit gathered data to the CDB located in the offline analysis centre. So locally, there are measurement, transmission, and control functionalities. The transmission issues are similar to the ones discussed in previous chapter. However, in this case the construction of the system node could be quite different.

## 3 LMN – Local Measurement Node

Local Measurement Node is the device that can perform measurements of several environmental parameters, store the results and send them to the central station. The method chosen for sending the data is radio transmission using ISM frequency. There are some protocols available for such a transmission. One of such protocols is ZigBee [3]. This protocol is becoming a very popular solution for short distance radio transmission and is widely used in sensor networks. Other possibility is to use modules offering raw data channel or channel with some low level modulation. Using such modules can be a good choice where nodes do not have to play role of routers for the network. In the proposed system nodes communicate directly with the central station so there is no need for implementing routing protocol.

The module chosen as the radio transmitter in the node is based on the CC1000 chip made by Texas Instruments (Chipcon). It can work with 433 or 868 MHz frequency depending on the version and can transmit data with the speed up to 78.6 kbit/s. It offers raw or encoded channel working in NRZ or Manchester mode. The radio module has the connection for external antenna and two interfaces to connect the microcontroller. One of them is the three wire synchronous serial interface used for controlling the mode and parameters of operation of radio module. The second serial interface is used for data transmission. It can work in synchronous or asynchronous mode depending on the CC1000 current mode of operation. As the main part of the measurement node the ATmega32 microcontroller has been chosen. Because the radio module works with 3.3 V of power supply, the low voltage version of microcontroller has been used in the system. The node can be supplied with several power sources. One possibility is the accumulator charged by the solar panel. Unfortunately, during long-lasting poor weather conditions solar energy wouldn't be sufficient so charging must be performed periodically by maintenance personnel. Such solution is uncomfortable and generates additional maintenance costs. Another possibility is to use high capacity batteries. An example of such battery is LS33600. It is the 3.3 V lithium battery of capacity 16500 mAh. Such battery has been used in the measurement node.

Because of supplying the node with the battery the special care has been taken on the power management. Both radio module and microcontroller offer many advanced power down and sleep modes. Radio module is switched into power down mode when not used. In this mode it consumes less than 1 $\mu$A of the current. Microcontroller is switched into power save mode and automatically waken up every second to perform measurements and store the results. After collecting 10–50 of data samples it switches the radio module on for sending data packet to central station. The number of data packets needed for performing the transmission can be set during node configuration.

The node has 4 16-bit analog and 8 digital inputs. Therefore, maximally there are $M = 4$ analog measures and 8-bit discrete vector, so maximally $m = 9$ bytes of useful data. Any unused input is configured as disabled for saving the battery power. Analog inputs can measure the voltage from any electronic sensor.

Digital inputs provide information about the state of switches or two-state sensors. The node can communicate with advanced sensors using serial interfaces. Both synchronous and asynchronous interfaces can be used. Asynchronous interface is compatible with RS485, synchronous interfaces are compatible with TWI and SPI protocols. Because many nodes send data to one central station, some protocol for accessing the medium must have been developed. The protocol is very simple. All transmission must be acknowledged. If the node does not receive the acknowledgement frame, it resends the data after 1 s. If such a situation occurs three times, it is the information about possible collision in the radio channel. While detecting the collision, the node randomly changes the time of next transmission. After five such changes the node turns into low frequency operation which means that it will perform measurements and transmissions every 30 s. If the node receives the acknowledgement frame, it returns to normal operation.

## 4 LPCN – Local Processing and Control Node

As a main unit of LPCN, the unaided field computer of PC or SBC (Single Board Computer) class can be used as well as a dedicated computer device can be designed. This computer works without any rotated parts and is closed in a special case resistant to weather conditions. The acquisition module and supporting coprocessor are joined as a PCI board or are a part of the device. The details are presented in Sect 4.2.

Two of the tasks presented previously could be realized in ways which guarantee better system performance related to computing power and power consumption. The first one is local processing. There is a necessity that some data sets have to be processed locally and this processing have to be performed in the fast and real-time way. The main goal for this is to produce appropriate control signals dedicated to local actuators, i.e. traffic lights, buzzers, automatic bars, etc. Taking into consideration the size of monitored areas, it is requested to install a relatively big number of LMNs. Let's assume k nodes for further considerations. So registration and processing a big amount of measures is necessary. Additionally, prediction algorithms used for local control are quite complex and they have time constraints, so there is a necessity to use powerful processing devices with low power energizing. The second task is a long distance transmission to CDB. Since data can be collected in a LPCN memory and local processing exists, the transmission between LPCN and CDB could be executed within a dynamic cycle, according to achieved results from local processing and a data buffer capacity. However, the transmission is considerably power-consuming. So there is a justification for finding out a method of reducing the power consumption related to radio activity time. The good answer for both problems above is FPGA technology [1,2]. Authors consider the use of this kind of circuit in order to use it in LPCN. There is also extra utility; it allows to update a hardware application many times according to changes in the coprocessor project.

The hardware solutions can be realized in various technologies executed in integrated circuits of the following types: ASIC, DSP, specialized processors, and also a programmable array. The hardware solutions as a rule allow to make calculations of one or even two orders of magnitude more quickly than the software solutions. The achieved speed of making calculations is evidently dependent on the method of the implementation and the applied technology.

Additionally, the application of a programmable array to the implementation of computational algorithms, in the relation to different technologies, enables us to change the hardware configuration of the programmable array. Consequently, one can exchange the working hardware application for the one realizing a different function or acting more quickly. The increase of the calculation speed can be held by making the calculations in parallel or executing the operation in the pipeline. Moreover, achieved results in the case of the pipeline solutions are comparable with the quickest solutions executed in the different hardware technology. However, utilization of the VHDL allows to apply the implementation of the hardware application in various programmable arrays. This mechanism is described in Sect. 4.3.

### 4.1 Remote Communication

An LPCN has to be a single, management and repository device connected locally with LMNs and remotely with a distant CDB. It has to be installed near measured regions, so local environment may be various but mostly not urban. The communication infrastructure in a given location is generally unknown. Thus, from the universality point of view, an LPCN has to service at least a few methods of distant communication. It can be a local area network via either cable or wireless connection, radioline based on radio modems, GPS/GPRS transmission, commutative or direct telecommunication line, etc.

The hardware and software platform described in Sect. 4.2 allows to use almost any communication solution dedicated to distant transmission. However, there is a point to creating a special supporting mechanism which is able to reduce time of transmission. The first reason is a time characteristic of the whole system running. The system has to work with time constraints, because depending on the results achieved from calculations the safety actions are locally performed, and remotely user is informed. The second one is related to power supply. If energizing source is not permanently available, power consumption is important because of battery lifetime. The radio, especially the transmitter, consumes more power when transmission is occurring then when it is in sleep mode. The last one is related to maintenance cost. In some solutions the cost of transmission depends on transmission time or data size (i.e. GSM/GPRS), especially when packet size of useful data is relatively large. All of this leads to the reduction of transmission time. If transmission speed is fixed, there is one simple way to diminish it. Useful data can be compressed and selected protocols should be simple, without control data overload. The FPGA technology mentioned previously gives an opportunity to use it also in order to compress data very fast and without significant increasing of power consumption. This mechanism is described in Sect. 4.4.

Finally, the distant transmission can be integrated with Internet, especially when CDB is connected to it. If the TCP/IP stack can also operate with an LPCN, the proper protocol set has to be used, mainly in transport layer. There are two ways. The UDP protocol uses less control data but transmission is unreliable and user has to control transmission from application layer. The TCP protocol delivers a reliable channel but adds more control data. In order to select appropriate solution, the analysis of channel has to be performed for each particular case.

### 4.2 Main Unit

In a prototype version, the main unit is a computer host station consisting of elements presented in Fig. 1 and based on any hardware supported by OS. To evaluate this version, the Virtex-5 FXT FPGA ML507 Evaluation Platform [11] with PCIE interface is used. The OS platform can be Linux. Linux seems to be the best choice. Linux is an open source, operating system, developed originally for home PCs. Today, Linux runs on a variety of CPUs including PowerPC, MIPS, Alpha, Sparc, M68000 and ARM. The most important criterion of an embedded system, such as in LPCN case, is timing constraints reliability.

From version 2.6, Linux kernel task can be preempted [8,9]. In the general-purpose operating systems, the process scheduler during execution of a system call is forbidden. This possibly leads to the situation when one task may block the processor until the system call returns, no matter how long that can take. From version 2.6 the Linux kernel, during the execution of system calls, periodically tests delays between the preemption point, and then the kernel task can be interrupted.

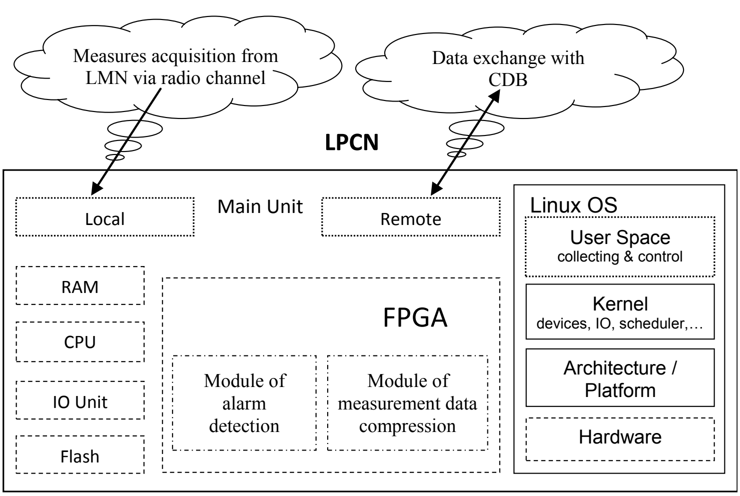



**Fig. 1.** The components of LPCN main unit

Embedded applications are customized to solve the problem in an original way. This requires customizing Linux to support the specific hardware design. It can be done easier when the system has the modular design with well defined components and API's which can be replaced or customized with minimal impact on other components. Figure 1 also shows the modularity architecture of the Linux 2.6 kernel design. Porting the system from different hardware is possible and convenient during moving the application from development to production platform.

One unquestionable advantage of the Linux is the scalability from the system that provide exceptionally large resources and high-throughput multiprocessor to the microcontrollers. Version 2.6 bringing microcontroller support into the Linux mainstream. It is done mostly by the adoption of the ţClinux distribution. The original ţClinux was a derivative of Linux 2.0 kernel intended for microcontrollers without Memory Management Units (MMUs). However, the Linux/Microcontroller Project has grown both in brand recognition and coverage of processor architectures. Today's ţClinux as an operating system includes Linux kernel releases for 2.0, 2.4 and 2.6 as well as a collection of user applications, libraries and tool chains. Thanks to Linux cross-platform mobility, after preparing the LPCN prototype and running it with the rest of the system, the LPCN as the SBC (i.e. TS7300 [10]) can be designed together with software and hardware projects prepared earlier.

### 4.3 Processing Support

The modules of alarm detection (Fig. 2) will begin the work after detection of signal informing about the end of collecting a the single package of the measuring data. This information is passed on to the suitable processing modules, which executes the operations on the current given measuring. The modules informing about the crossing of the threshold value for the given measurement will be the first processing modules. The threshold value should be so well-chosen to be able to affirm the possibilities of the pronouncement of the threat. On the example, one can define the threshold value of the pressure of water in soil, from level on which sliding of the ground can occur.

The second processing module informs about crossing the threshold value of differences between $K$ of next measurements for the given sensor. In effect, one can notice violent growth of the signal for the given measuring sensor and inform that the considerable change of the measurement happened in the given place in the short period of time. The last processing module informs about crossing the threshold value of differences between $L$ of measurements from the given class of measurements. In effect, one can get information about the difference of the value of measuring signals from all measuring sensors from the given class. This information will be essential for generating the spatial presentment of the value of measuring signals in the given time. This will let define in what area the threat may happen. I.e. in which place and what size a slope may slide.

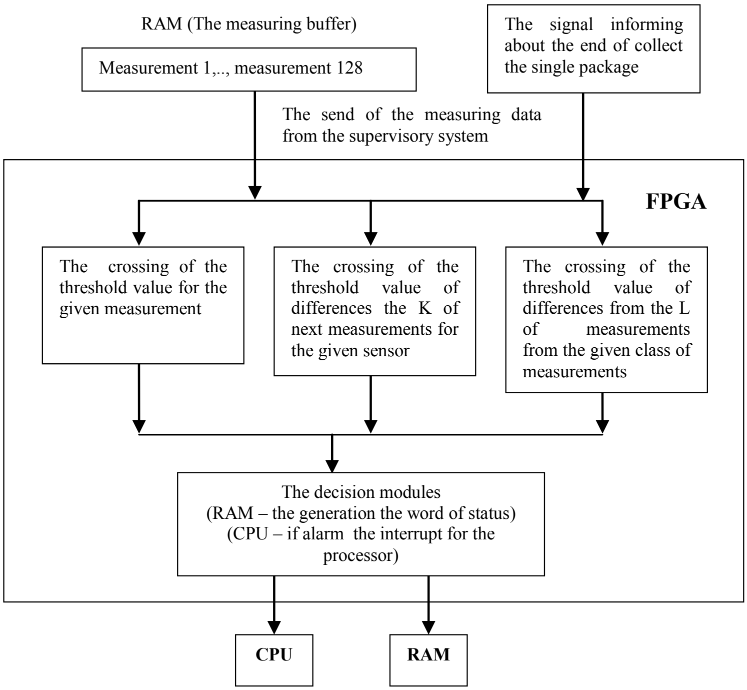



**Fig. 2.** The modules of alarm detection

As a result of execution of the modules is generation a status word which informs about alarm pronunciation for a given measurement sensor as well as, in case of triggering alarm, generation an interrupt for processor.

### 4.4 Data Compression

There are several streams of data to send over the net in the measuring system. The data is acquired by sensors (LMN) and transmitted without compression to a host station (LPCN). In the host station the data is buffered, processed and transmitted to the main station (CDB). Before the transmission the data should be compressed using lossless compression method. The gain of that compression is reducing the amount of the data sent, without loss of any information. It will reduce the time of the radio activity, the consumed power, and save its batteries.

Let us assume for example, that there are $k = 32$ sensors, each of them acquires $M = 4$ measured values per second using 16-bit resolution (so $m = 64$ bytes). This gives stream of data $k \times m = 32 \times 4 \times 16 = 2$ kbits/s. This stream is buffered in a host station and should be sent to the main station CDB, using one transmission process per hour. It gives 7 Mbits of data to be compressed for one compression process, so the off-line compression algorithm may be used with the best possible compression factor.

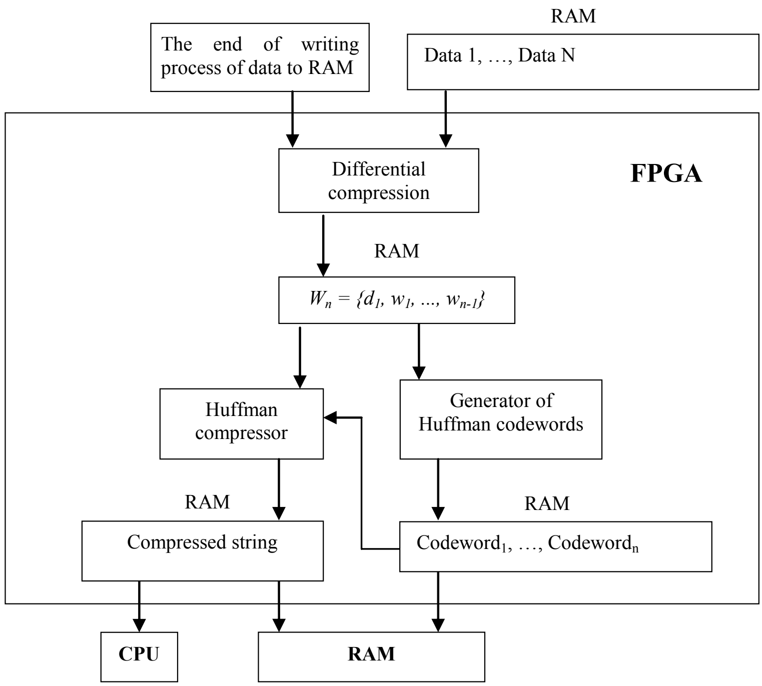



**Fig. 3.** The scheme of compression in FPGA structure

The compression algorithm must be implemented in FPGA, so it should be simple and not use a lot of memory. The compression process begins when the compression module receives the signal of completing the data in RAM. The first step of the compression process is to separate streams of data containing particular measured values. Those values are physical parameters, for example temperature, pressure, precipitation, measured periodically, so the consecutive values of each parameter are self-similar.

Motivated by self-similarity property, the next step is using differential coding method for decorrelation of the data for every stream. For the given data string $D_n = d_1, \ldots, d_n$, the difference between previous and current value is calculated, for each value. The result is string $W_n = d_1, w_1, \ldots, w_{n-1}$, where $w_i = d_{i+1} - d_i$.

The next step of the compression process is statistical coding of string $W_n$. For simplicity it can be the Huffman algorithm in off-line version [6,7]. The generator of Huffman codewords makes and writes to the RAM the book of codewords. Huffman compressor takes symbols from $W_n$ string and codes them using codewords from RAM. The compressed string is written to RAM and waits for sending. The scheme of compression in FPGA structure is presented in Fig. 3.

In the case of emergency situation, the data should be sent immediately without buffering and, furthermore, the data may not be in strong correlation. The solution for the compression in that case is to use some kind of the fast on-line compression algorithm or send the emergency data without any compression.

## 5 Conclusions

The presented idea of a distributed computer system is only a concept. However, authors have wanted to prove that currently available IT techniques allow to design and build a comprehensive system, applicable to the area mentioned above. This article can be a good study for designers. Additionally, in order to put the idea into effect, more detailed projects and research have to be done. All of the authors have experience with issues regarding parts of the described system.

The presented solution is quite cheap, based on common components and some interesting improvements are involved. Using the FPGA technology gives a possibility to build a universal, multiplatform coprocessor and achieve high speed computing with low power consumption. Specialized compression can be used also between measurement nodes and the host station. However, nodes should be able to support it. The quite important advantage is that the system can be integrated with existing devices coming from various producers, both on a measuring and on a computing level.

## References

1. Ziebinski, A., Sroka, W.: Realizacja funkcji statystycznych w sterowniku przemyslowym z wykorzystaniem ukladu FPGA. In: Mazur, Z., Huzar, Z. (eds.) Modele i zastosowania systemow czasu rzeczywistego, WKL (2008)
2. Ziebinski, A., Znamirowski, L., Sroka, W.: Implementacja wybranych funkcji sterownika przemyslowego w ukladzie programowalnym. In: Systemy Czasu Rzeczywistego, Metody i zastosowania. WKL (2007)
3. ZigBee Proffessional Guide. A Webcom Publication (2008)
4. Gaj, P.: Zdalne monitorowanie obiektow z wykorzystaniem sieci komorkowych. In: Napedy i Sterownie. No 2(94), Poland, Drukart, pp. 50–52 (2007) ISNN 1507-7764
5. Gaj, P., Holodok, A., Bigewski, Z., Kwiecien, A.: Zdalny monitoring z wykorzystaniem sieci GSM. In: Techniczne i spoleczne problemy z zastosowania Internetu, Poland, WKL (2005)
6. Huffman, D.: A method for the construction of minimum redundancy codes. Proceedings of the Institute of Radio Engineers 40, 1098–1101 (1952)
7. Hankamer, M.: A modified Huffman procedure with reduced memory requirement. IEEE Transactions on Communication 27, 930–932 (1979)
8. Aseem, R.: Linux Kernel 2.6: The Future of Embedded Computing. Part I (2004), http://www.linuxjournal.com/article/7477
9. Linux 2.6 for Embedded Systems, closing in on real time, http://www.lynuxworks.com
10. TS7300 manual rev. 1.5, Technologic Systems. USA (2008)
11. Virtex-5 FXT FPGA ML507 Evaluation Platform, http://www.xilinx.com

# The IEEE Wireless Standards as an Infrastructure of Smart Home Network

Mateusz Grabowski and Grzegorz Dziwoki

Silesian University of Technology, Institute of Electronics
grzegorz.dziwoki@polsl.pl

**Abstract.** Home environment management systems, also known as smart home systems, become more and more popular subject in recent days. Such systems can be a support for elder or handicapped people as well as can save energy and decrease pollution emission. Although dedicated systems have been already presented on the market they are rather expensive and require major interference in existing building infrastructure. Therefore easy to use solutions are strongly demanded by modern home owners. The paper presents general requirements that should be fulfilled by the network of a smart home system and discusses some aspects of the application of wireless networks for that purpose. The main attention is put on the network operation on the level of the physical layer. The simulation and measurement results are also presented in order to evaluate usefulness of the proposed wireless approach.

## 1 Introduction

Smart home system can be defined in many ways, depending on the area of problem exploration. One of those definitions, appropriate from the network infrastructure point of view, was formulated by the Intertek, and reads as follows [1]:

*"Smart home is a dwelling incorporating a communications network that connects the key electrical appliances and services, and allows them to be remotely controlled, monitored or accessed".*

Smart home system infrastructure consists of a residential gateway, a control network and a media network as it is presented in Fig. 1. The control network interconnects controllers, sensors and actuators for control and security information exchange, the media network is in charge of audio and video transmission, whilst the residential gateway is responsible for joining them to outer services like e.g. the Internet, the GSM network etc. The networks can be wired or wireless as well as specifically standardized for home applications (EIB/KNX, X10) or adapted for that purpose like the IEEE open standards. The latter are widely popularized and cost efficient but wired one are difficult to implement in existing buildings without their severe modernization.

This paper analyzes functionality of the smart home network based on the wireless IEEE standards. The network requirements, defined in the next section, help to evaluate the proposed approach. A comprehensive characterization of

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 302–309, 2009.

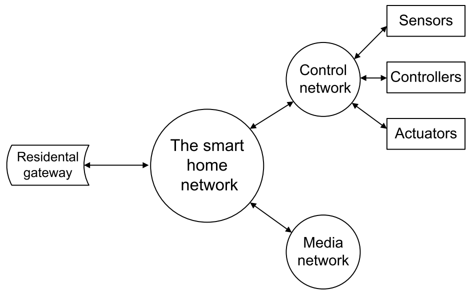



**Fig. 1.** The network structure of smart home system

the smart network is extensive issue and here it is limited to an analysis of the primary physical parameters like the transmission range, energy consumption, interferences and their influence on information rate. The considerations are supported by the simulations and measurements experiments.

## 2 The Smart Home Network Requirements

The network infrastructure used for smart home construction should satisfy the following parameters [2]:

1. **Accessibility** – access to the network should be provided in the whole house. In this paper for simulation purposes it is assumed that the house area is about $225\,\mathrm{m}^2$ (15m x 15m).
2. **Transmission rate** – the network must support rates enough for various types of services (e.g. command, media transmission etc.). The transmission rates vary from a few kbit/s up to 27 Mb/s for HDTV (High Definition Television) or only 4–7 Mb/s for SDTV(Standard Definition Television).
3. **Reliability** – the entire system has to work continuously. Security and fire networks are very often connected to fire or police stations, hence they should not give false alarms.
4. **Availability** – the system components should be easily available on the market to avoid situation when some unique products (devices) are used.
5. **Safety** – transmission should be coded to protect against eavesdropping and hacking.
6. **Flexibility** – the network solution must be universal so that it would be possible to implement such network in every house. It should be easy to modify and configure such network.
7. **Maintenance** – the system should be as maintenance free as possible. For instance if the sensors were supplied with batteries, they should work at least one year.

Above mentioned requirements have to coexist with the quality parameters demanded by end users. For example some QoS (Quality of Service) factors, which have strong impact on service realization are defined in the ITU-T G.1010 standard. One of them, the bandwidth, has been already mentioned in the above list as transmission rate. The next relevant parameters are as follows [3]:

1. **Delay** – time taken to establish service, from user's request to reception requested response.
2. **One-way delay** – time needed for a packet to reach its destination.
3. **Delay variation (jitter)** – difference between packets arrival times (caused by queuing) Jitter reduction is performed by buffering and internal delays.
4. **Information loss** – each application has a limit of lost packets or bits. If this limit is exceeded then information cannot be decoded correctly.

The presented conditions have meaning of guidelines for design and analysis of smart home network solutions. But their definitions give only some limited insight into the discussed problem, which is especially helpful for the system evaluation on the lowest, physical layer. The complete characterization should take account of different network layers demands and peculiarities. It is necessary to notice that fulfillment all those requirements on different network layers is very complex task. Therefore in this paper the attention is mainly put on primary transmission problems over wireless media.

## 3 The Networks Selection

Smart building needs two types of the network functionalities in order to realize control and media purposes, as it was presented in Fig. 1. From the integration and the maintenance viewpoint it would be desirable that only one network infrastructure carried out all necessary tasks. Connection to the outer network, like Internet, points to use of the IP network. However, the possible integration on higher system level does not always mean the same opportunity in the lowest layers, especially on the physical one.

Taking into account FLEXIBILITY requirement, a wireless medium seems to be a correct choice for the whole smart network infrastructure, because it requires only a minor influence on building structure. Moreover, almost all new devices like palmtops, laptops, cameras, etc. are equipped with the wireless standards, for instance Bluetooth or IEEE 802.11b/g. But hostile nature of the wireless environment (channel fluctuations, interferences) requires knowledge about the propagation characteristics in order for suitable placements of the network nodes. Those considerations are important from ACCESSIBILITY viewpoint. On the other hand the obvious differences between media and control transmission justify a distinct approach to design these specific networks.

### 3.1 High Throughput Network

The network for media purposes does not have to cover the whole house area. Building corners, specific rooms etc. can be out of range of the media services such as high quality media streaming (voice and video transmission) and the broadband Internet access. The most demanding HD (High Definition) video transmission requires data rate at least 27 Mb/s.

The following standards can meet those demands: IEEE 802.15.3 [4], IEEE 802.11 [5], ECMA-368 [6]. The first one works in the ISM band (Industrial, Scientific, Medical), but it is the least popular although it has built in some QoS capabilities in the MAC layer and was designed for streaming media purposes. Lack of the devices supporting that standard contradicts AVAILABILITY condition. Another IEEE specification, IEEE 802.11, although more suitable for asynchronous access (Internet), can be also adapted for streaming media. Possibility of the operation in 5 GHz band is the additional advantage. Also ECMA-368 adopted by WiMedia Aliance for the UWB (UltraWide Band) transmission does not work in the overcrowded ISM band and can utilize the target spectrum from 3.1 GHz to 10.3 GHz, and supports the transmission rates up to 480 Mb/s. It is a promising approach but currently UWB equipment can be bought mainly on the US market and they operate in the lowest bands of the UWB spectrum. For European users it is not too useful solution, especially as on the basic of the European Union commission decision that frequency band is permitted for the UWB transmission only until December 2010. Concluding, IEEE 802.11, also known as WiFi (Wireless Fidelity), seems to be the interesting choice for the physical layer of the *High Throughput Network* at present.

### 3.2 Low Throughput Network

What about control network? Is not WiFi suitable choice for it? What are necessary data rates? Smart home can be equipped with many sensors like temperature, noise level, presence, fire, security, light, humidity etc. The number of sensors can vary from a few up to a few hundreds. Their number depends on a system complexity, functionality etc. According to QoS, low data rates with high reliability and 100% coverage are sufficient for control purposes. Application of IEEE 802.11 for the control network can ensure those requirements but all the network nodes have to be supplied with energy from the electrical grid. It is highly disadvantageous owing to energy consumption as well as lack of the clear differences to wired control network. Those reasons preclude use of the WiFi infrastructure to exchange control information. But among the IEEE standards there is powerful candidate marked as IEEE 802.15.4. It belongs to WPAN (Wireless Personal Area Networks) alike well known Bluetooth (also adopted as the IEEE 802.15.1). But as opposed to Bluetooth, IEEE 802.15.4 has much better energy management and can support network with dozens of nodes [7].

The ZigBee specification is the extension of the IEEE 802.15.4 PHY (PHYsical) and MAC (Media Access Control)layers. Two new layers was added: Network/Security Level and Application Framework [7]. This type of network is suggested to form *Low Throughput Network* in smart home. Zigbee devices should work in so called beacon mode because in that case they know the exact transmission time and can be put into power down mode. The power saving mechanism can prolog battery life up to several years, which makes nodes almost MAINTENANCE free.

ZigBee network topology permits data and control messages to be send between nodes through multiple paths. It allows to achieve a high connection RELIABILITY and to build networks on the larger scale (ACCESSIBILITY). Additionally ZigBee offers a access control lists and 128 bit encryption which ensure a SAFE transmission.

## 4 Analyses of the PHY Layers

The section discusses the simulations and measurements results provided to evaluate to what extent working conditions of the above selected standards meet smart home requirements. Some parameters of the wireless communication link and the energy consumption of the low rate devices were examined. The obtained results are the average over 100 second of the simulation run and one measurement trial.

### 4.1 Transmission Range and Data Rates

Received power and that follows transmission range are strictly connected with used frequency band. Both, the high and low throughput networks can work in the ISM band, and from signal propagation viewpoint they operate similarly.

The link simulations are performed using the ns-2 network simulator and next they were compared with the measurements of WiFi network, which were done for two different APs (Access Point) 3com and Edimax and the building layout presented in Fig. 2. The transmitted power equaled 16 dBm. The average received power values were obtained by the Network Stumbler, the software run on the personal computer.

The power measurements were done in the building where walls were made from bricks. In order to explore the worst situation (the largest distance between two WiFi node in the building) the AP is in the corner and the received power of the client device was measured in the points placed on diagonal (the circle marks in Fig. 2). The results are presented in Fig. 3. In the same figure the simulation results for shadowing model [8] with the path loss exponential $\beta = 4$ is also depicted.

The $\beta$ value was selected in order to represent the worst case of the measurement results.

According to the IEEE 802.11g specification the raw rate of 54 Mb/s is achieved for the received power at least $-70$ dBm, whilst the lowest rate for $-86$ dBm [5]. It necessary to notice that the raw rate on PHY layer does not include influence of packet retransmission, IP and TCP headers, etc. on effective information rate.

Table 1 presents the effective information rates for TCP and UDP protocols if the maximum raw data rate is possible. It can be noticed that enabling handshake in MAC layer reduces achievable rates in TCP by 11% and in UDP by 17%.

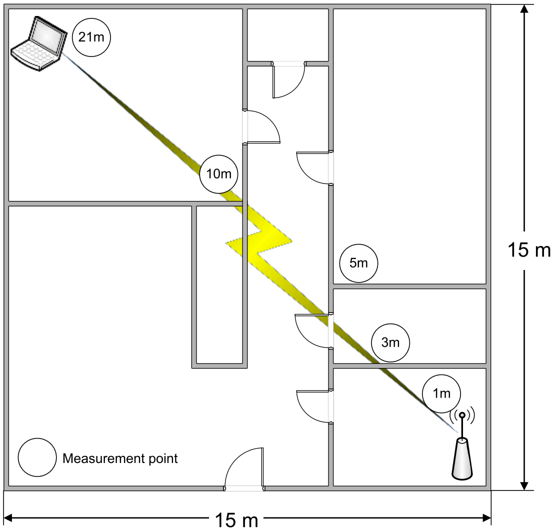



**Fig. 2.** The measurement scenario and the building layout

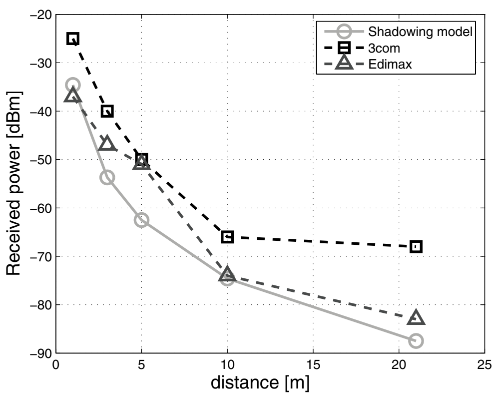



**Fig. 3.** The received power vs. distance from the AP

**Table 1.** Transmission data rates

| | Measured TCP | Measured UDP | Simulated TCP | Simulated UDP |
|---|---|---|---|---|
| RTS/CTS off | 21.1 Mb/s | 22.2 Mb/s | 22.3 Mb/s | 31.0 Mb/s |
| RTS/CTS on | 20.9 Mb/s | 22.1 Mb/s | 19.8 Mb/s | 25.6 Mb/s |

However, the real measurements do not confirm that. Although the simulations provided by ns-2 show that UDP transport protocol can support transmission of the HDTV signal (27 Mb/s), the measurements prove that such high rates are out of reach in the real environment. The only way is to use better video coding with lower resolution.

### 4.2 Interferences

Due to the fact that APs can operate on adjacent channels, co-channel interferences can occur. Also transmissions on the same channel occur very often, because in most cases the devices use default assigned channel. The measurement carried out in the suburbs of Warsaw confirmed this rule. For 16 available networks, only 4 used different channel number other than no. 11. In case of sharing the same channel the networks performance can strongly decrease. Using ns-2 it is possible to check what happens if two APs start using the same channel. The simulation scenario consists of two separate pair of nodes that are able to interfere each other. The distance between nodes in the given pair would allowed to reach maximum data rate of 22.4 Mb/s if neighbouring transmission were not presented. Otherwise the effective information rate is only 10.3 Mb/s.

Distortion from home appliances creates the additional problem. Especially microwave ovens working in the ISM band are the most problematic. The results in Table 2 show their influence on information transmission. The measurement scenario assumed 1 m separation of the wireless devices but 2 m far from the client, the microwave oven was switched on. Data rates were measured for three cases, in each case the devices operated on the different channel. Among examined channels, transmission on channel no. 6 was the most sensitive to microwave oven distortion. Channel 13 is the most resistant to these interferences. The results show that dynamic channel assignment could be a solution to increase of the network performance.

**Table 2.** Average information data rate in the presence of microwave oven interference

| Channel no 1 | Channel no 6 | Channel no 13 |
|---|---|---|
| 15.6 Mb/s | 8.75 Mb/s | 16.1 Mb/s |

### 4.3 Energy Consumption

In the case of the low throughput networks with even dozens of nodes the problem of energy consumption becomes crucial especially if they are supplied with batteries. Simulation shows the possibility of energy adjustment when devices working in beacon-enable mode in star topology with one AP (only connected to mains). For small home areas it is useful and simple topology [2]. Table 3 presents maximum time in which one node can have constant connections under certain superframe settings (simulation time equaled 3600 second).

The raw transmission rate was set to maximum 250 kb/s. The superframe parameters, which indicate among others the period of time between two consecutive superframes and also time when nodes are in power down mode, are the reason of reduction of the effective data rate. Simulation shows that with suitable configuration parameters, constant connection can last almost one year and a half.

**Table 3.** Average information data rate vs. predicted battery lifetime

| Data rate | Lifetime |
|---|---|
| 3.97 kb/s | 2866 hours |
| 0.98 kb/s | 11476 hours |

## 5 Conclusions

Few years ago a computer implemented in a car was something new, extraordinary. Nowadays, it is the common practice. The same situation can happen with houses. But unfortunately, today progress in the smart home field is slowed down by high costs. The answer can be a solution based on available universal standards just like the IEEE 802.15.4 and the IEEE 802.11g.

This paper defines requirements that smart home network should meet and tries to answer on the base of the simulation and measurements how wireless standards meet them from physical layer viewpoint. Streaming media can work in some extent under proposed high throughput network but it is suggested to separate them form Internet connection distributed in home by using another AP, which works on different channel. The use of UWB transmission could be a solution but currently it is less popular. Interference are another crucial problem because all the networks work in the same ISM band.

The results of the considerations show that wireless smart home network can be promising approach, which is especially important for existing buildings. Still a lot of questions remain without answer therefore further investigations (with extensive statistical analysis) are strongly needed to determine the best configurations parameters.

## References

1. King, N.: Smart Home – a definition. Intertek Research & Testing Centre (2003)
2. Grabowski, M.: Application of modern network solutions for smart home construction. MSc dissertation, Gliwice (2008)
3. End-user multimedia QoS categories. ITU-T Recommendation G.1010 (2001)
4. Wireless PHY and MAC Specifications for High Rate WPANs. IEEE Std 802.15.3 (2003)
5. Wireless LAN Specificatons family. IEEE Std 802.11
6. High Rate UWB PHY and MAC Standard. ECMA-368 Standard, 2nd edn. (2007)
7. Kinney, P.: ZigBee Technology: Wireless Control that Simply Works, White Paper (2003)
8. Rappaport, T.S.: Wireless communications, principles and pratice. Prentice-Hall, Englewood Cliffs (1996)

# The Graphic Representation of Areas of Rough Sets Characterizing the Groups of Scheduled Tasks

Henryk Piech

Czestochowa University of Technology, Dabrowskiego 73
h.piech@adm.pcz.czest.pl

**Abstract.** The graphic conception of description of scheduling question with help of rough sets give us the possibility of looking on the mutual, relative location the parameters of tasks. It is the way on synthesis of results got with help of different algorithms scheduling also. Thanks to the graphic illustrations we can easily emphasize the general feature of group of scheduling tasks and assigned to the processors, such the bottom level of positive area, border and upper level of zone of upper probability. Graphic illustration of probabilities of conception takes into account the effects of integration of results ranking got for file of algorithms simultaneously.

## 1 Introduction

The problem of scheduling tasks has the wide uses both in computer science in distribution of tasks in multiprocessor structures [6] and in connected with optimization of organization questions [5], the sequent decision, the management the complex enterprises [11] , the recourses [1] and the projects. The practical utilization the methods of scheduling tasks reaches the questions of organization of production, systems of mass service ( the use in management the arrivals and take-offs of airplanes [2]), the management the realization of investment [3], the assignment of recourses in projects [2], the packing the boxes [4], the ranking on parallel machines the problems etc. The utilization disjunction network [9] and determine methodologies have usually combinatorial character. Polynomial method hardly ever give exact or optimal solution . In many works was presented discussion and analysis of efficiency end complexity of scheduling tasks methods. Conception depends on synthesis of results of categorization with help of chosen algorithms to scheduling independent tasks [10]. We should treat this kind of date as needing additional analysis taking into account interval, fuzzy or probabilistic its character [7]. The probabilistic analysis can be treated as the most objective, because it depends exclusively only of degree of utilization of data, but not experts' subjective opinions [8]. The graphic conception of description of scheduling question with help of rough sets give the possibility of looking on the mutual, relative location the parameters of tasks. This is the way on synthesis of results got with help of different scheduling algorithms. Thanks to the graphic

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 310–317, 2009.

illustrations we can more easily emphasize the general feature of scheduled and assignment to processors task, such the bottom level of positive area, border and upper level of zone of upper probability.

## 2 The Conception of Structuralization of Rough Sets in Reference to Tasks

We are far from description of theory of rough sets [11] but want to come up to its describing the areas of probabilities part. Three areas are specified : positive zone $Pos(O)$ (bottom probability $Lower(O)$), negative zone $Upper(O)$ (upper probability $Neg(O)$) and border $Bnd(O)$:

$$\begin{aligned} Lower(O) &= Pos(O) = \bigcup_{e(i)\subseteq O} \{e(i)\} \\ Upper(O) &= \bigcup_{e(i)\cap O\neq 0} \{e(i)\} \\ Neg(O) &= O - Pos(O) \\ Bnd(O) &= Upper(O) - Lower(O) \end{aligned} \tag{1}$$

where:

$O$ – set of objects
$e(i)$ – $i$-th object

The graphic interpretation of areas of objects would can for example look as in Fig. 1.

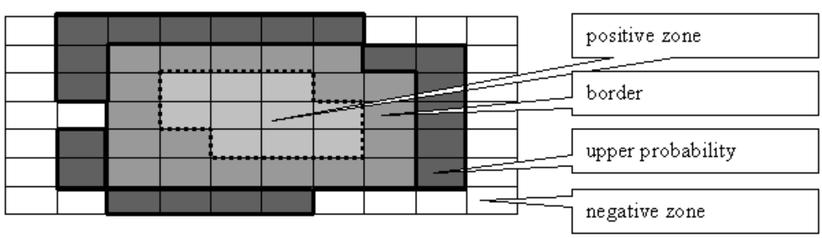



**Fig. 1.** Zones (areas) of probability of objects sets

In situation, when the tasks are the objects, we want to assign to individual processors (to scheduling), graphic scene of such objects we can divide on individual processors and consider the order of assignment of tasks in following way (Fig. 2).

The majority of effective algorithms of scheduling are based on not growing ordered times distances of realization tasks. Processors play less essential role,

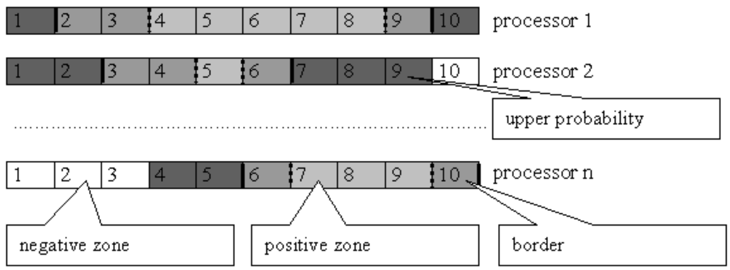



**Fig. 2.** Probability of tasks sets in processors-tasks space

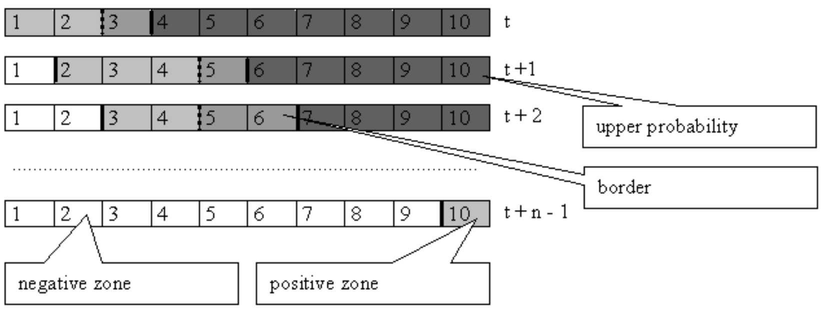



**Fig. 3.** Graphic way of registration the areas of probability of realization of next assignments (introduced in dynamic spread in time of reorganization of structure areas form), $n$ – number of tasks, $t, t+1, t+2, \ldots$ – moments of assignment of next tasks

final role rely on define sequence of choice tasks from the list [6]. In such situation the areas (zones) respond to probabilities of realization the conception "assign task" we can put on rank list. We can do it in dynamic, changing in time, version of realization next tasks assignments (Fig. 3) or in static version by simultaneous creating all areas for individual assignments (Fig. 4 and Fig. 5).

The next phase of assigning the tasks with list were presented on Fig. 3. For every of phase was showed the full structure of areas of probability of realization conception "assign task" in reference to tasks with lists. The next phases were marked by sequence of moments of assigning problems: $t, t+1, t+2 \ldots$. All areas on Fig. 3 were marked, with negative area inclusive, which begins appearing after assigning first problem and expand with the moment of distribution of next tasks. Only positive area after assigning the last problem stays obviously.

On Fig. 4 were left only positive areas but they are distributed only on full basic list. However not always it is successful in such way unambiguously to qualify individual ranges, which overlap on each other and can to be also disjoint (its depends from which algorithm we are reaching to list's elements). It will be the more precise way the graphic system of description, which give us possibility to refer to every position from list by fourfold structure of areas of probability (Fig. 5).

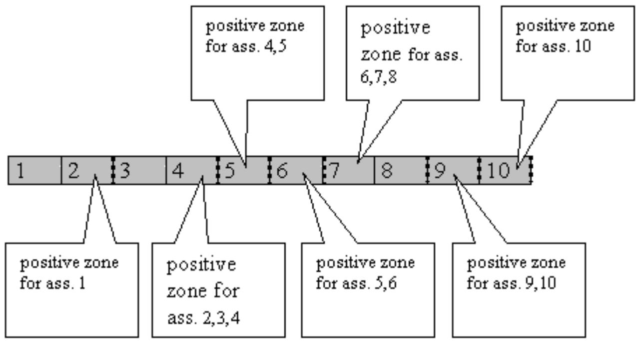



**Fig. 4.** Distribution of positive areas on base list for successive assignments

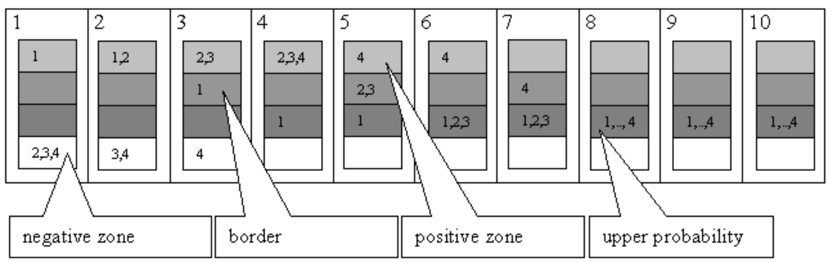



**Fig. 5.** Distribution of areas for successive four conceptions "assign task". Succeeding four rectangles refer to characteristic areas.

## 3 The Integration of Results from Different Algorithms in Reference to of Rough Sets Structuralization

We dispose, in majority of algorithms of scheduling the independent tasks, fundamentally one not growing ordered list of tasks according to times of realization [8]. If every of chosen algorithms lets different sequent of assigning problems we can use with categorization leaning on synthesis of those results. For example if we will use four algorithms of scheduling and with their help we will get several sequents of ranking we can obtain the synthesis sequent on base of categorization. Instead of categories for concrete task we can add up their participation in sets of proposal till given moment of assignment. To graphical present the effects of synthesis the results of scheduling with utility four algorithms for individual assignments, with showing the probability of conception of choice the task, we can rescale these results in reference to maximum added up categories in current stage of assigning the tasks (number of assignment) (Table 1).

We can now attempt to create graphic method for qualifying the individual zones for all assignments of tasks (Fig. 6).

**Table 1.** Example-probability of conception "assign the task" ($zi$) in next stages of distribution of tasks: $p(zi, np)$; where $np$ – number of assignment

| no assignm. | z1 | z2 | z3 | z4 | z5 | z6 | z7 | z8 | z9 | z10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 1.000 |
| 2 | 0.250 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.750 | 0.000 |
| 3 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 1.000 | 0.000 | 0.000 |
| 4 | 0.432 | 0.189 | 0.000 | 0.000 | 0.000 | 0.000 | 0.378 | 0.000 | 0.000 | 0.000 |
| 5 | 0.000 | 0.684 | 0.000 | 0.000 | 0.000 | 0.316 | 0.000 | 0.000 | 0.000 | 0.000 |
| 6 | 0.000 | 0.000 | 0.667 | 0.000 | 0.333 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| 7 | 0.000 | 0.000 | 0.000 | 0.118 | 0.265 | 0.000 | 0.618 | 0.000 | 0.000 | 0.000 |
| 8 | 0.000 | 0.000 | 0.000 | 0.526 | 0.000 | 0.474 | 0.000 | 0.000 | 0.000 | 0.000 |
| 9 | 0.000 | 0.000 | 0.000 | 0.000 | 0.500 | 0.500 | 0.000 | 0.000 | 0.000 | 0.000 |
| 10 | 0.000 | 0.000 | 0.000 | 0.000 | 1.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |

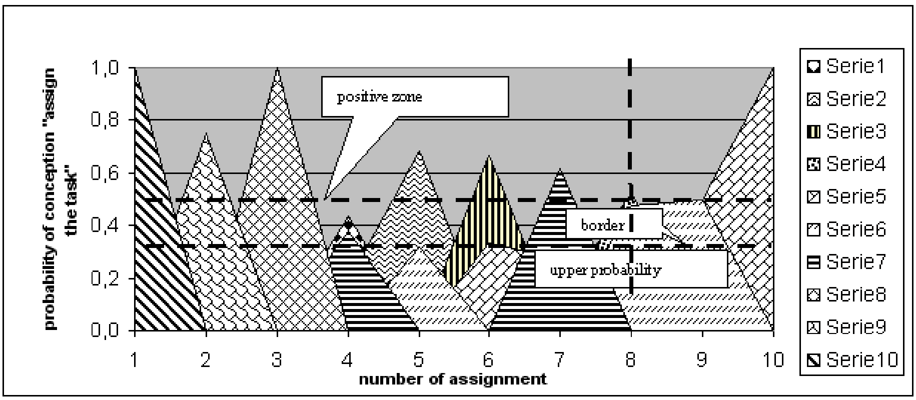



**Fig. 6.** Effects of synthesis of results of four algorithms of scheduling according to assignments

The graphic illustration of division on areas of probabilities of assignment the tasks introduced in amassed form (for all stages of assignment of tasks) is not always unambiguously in relation to effects of choice of tasks. On Fig. 6 number of series responds to the number of task. For example the eight assignment to positive area we will classify both the problem $z4$ and $z6$ ($p(z4, 8) > br\_ar\_pos = 0.4$ and $p(z6, 8) > br\_ar\_pos = 0.4$; where $br\_ar\_pos$ – low border of positive area). We can also present tasks, which were assigned to group of processors (Fig. 7). In this case we can also to define the zones of probability, and next series refer to next assignments directly.

As a low border of positive area we can mark level under which contain all settlement relating to the choice of conception "assign task" characterizing with maximum value of probability. It is visible on Fig. 6 and Fig. 7.

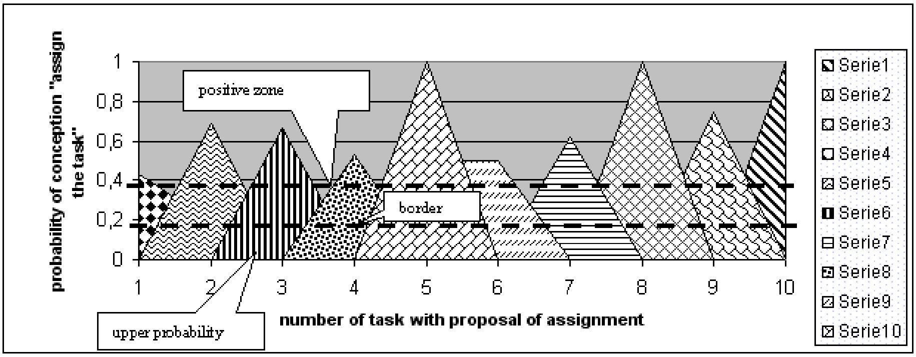



**Fig. 7.** Effects of synthesis of results scheduling by 4 algorithms according to tasks

$$br\_ar\_pos = (entier(n * \min_{1 \le np \le lp} \{ \max_{1 \le i \le n} p(z\_i, np)\}))/n$$
$$= entier(4.32)/10 = 0.4 \quad (2)$$

where:

*entier* – after cuting of fractional part,
$np$ – number of assignment,
$i$ – task number,
$lp = n$ – the number of assignments levels equals the number of tasks.

As upper border of upper probability we can mark the level of probability which exceed the lowest quotations of categorization but not more than on conventional graduation $(1/n)$:

$$br\_prob\_up = (entier(n * \min_{1 \le np \le lp} \{ \max_{1 \le i \le n} p(z\_i, np)\} + 1))/n$$
$$= entier(2.18)/10 = 0.2 \quad (3)$$

## 4 The Algorithms of Categorization of Data and Results Characterizing the Scheduled Tasks (General Presentation)

We prepare date for algorithms of scheduling on base of uncertain information about times of their realization. For example for ten-degree scale it can be following set of time categories of tasks:

$z1$ estimated time $t(z1) = 3.32$u.t. time category $k_t(z1) = 4$
$z2$ estimated time $t(z1) = 4.92$u.t. time category $k_t(z1) = 5$
$z3$ estimated time $t(z1) = 9.25$u.t. time category $k_t(z1) = 10$
$z4$ estimated time $t(z1) = 0.66$u.t. time category $k_t(z1) = 1$
$z5$ estimated time $t(z1) = 1.12$u.t. time category $k_t(z1) = 2$
$z6$ estimated time $t(z1) = 7.81$u.t. time category $k_t(z1) = 8$
$z7$ estimated time $t(z1) = 2.51$u.t. time category $k_t(z1) = 3$

**Table 2.** The list according to category of time of realization tasks

| algorithm CMM | |
|---|---|
| 3 | z k_t(z 3) = 10 |
| 6 | z k_t(z 6) = 8 |
| 2 | z k_t(z 2) = 5 |
| 1 | z k_t(z 1) = 4 |
| 7 | z k_t(z 7) = 3 |
| 5 | z k_t(z 5) = 2 |
| 4 | z k_t(z 4) = 1 |

The not growing ordered list of tasks for algorithm CMM [5,7] looks as in Table 2.

In categorization algorithm, moving along the ranges of categories (in direction from the highest category to the bottom) we study, which from estimated (in approximate way) times of realization of tasks were founded in zone of current category. The code or number of this category will be the determinant of category the tasks, which were found in this zone. Categorized and well ordered the list of problems can be used by different algorithms of scheduling in different way [11]. The next stage is the integration of results got with help of different algorithms. General algorithm of such integration consist on estimating the integrated weight (on base of weight component refer to successive assignments of tasks with omission the tasks, which were assigned already). The methods of getting of integrated weight can be diverse also, which in consequence leads to different levels of probability of conception "assign task". In algorithm of integration of results categorization tasks by la algorithms for every assignment we accumulate weight for every category, which be considered in studied interval. We skip categories tasks obviously, which were already assigned. We choose maximum accumulated weight $r(i)$ and we designate to assignment task which responds to this weight z $r(i)^{-1}$. This problem receives status assigned. Next algorithm will estimate the probabilities of conception "assign task" as well as it defined areas of probabilities of this conception (in generalized on all assignments).Presented algorithms use with widened information about chosen problems by group of algorithms. In this case qualification "widened" means respond to utilization by algorithms all tasks and not only these chosen. We read in accumulated category for every assignment and every problem. We make the standardization the accumulated categories in every stage of assignment in reference to the sum of all accumulated categories then. Maximum accumulated category in concrete assignment indicate on choice of conception "assign task". The minimum value is from among maximum accumulated categories the bottom border of positive area for all assignments.

## 5 Conclusions

Graphic illustration basing on described in matrix and vectorial structures data permits on global estimation the ranges of probability of conception individual assignments. Graphic illustration of probabilities of conception takes into account the effects of integration of results ranking got for file of algorithms simultaneously. These effects were we can present in reference to numbers of assignments or in reference to codes of concrete tasks. The global treating the borders of areas is the effective variant of classify the concrete conception "assign task" as preferred or rejected groups of scheduling problems.

## References


1. Błażewicz, J., Lenstra, J.K., Rinnooy Kan, A.H.G.: Scheduling subject to resource constrains: Classification and complexity. Discrete Appl. Math. 5, 11–24 (1983)
2. Elmaghraby, S.E. (ed.): Symposium on the Theory of Scheduling. Springer, Berlin (1973)
3. Falk, J.E., Horowitz, J.L.: Critical Path problems with concave cost time curves. Menagement Sci. 19, 446–455 (1972)
4. Grey, M.R., Johnson, D.S.: Approximation algorithms for the bin packing problems. In: Analysis and Design of Algorithms in Combinatorial, pp. 147–172. Springer, Heidelberg (1981)
5. Jackson, J.R.: An extension of Johnson's results on job lot scheduling. Naval Res. Logist. Quart. 3, 201–203 (1956)
6. Langston, M.A.: Improved LPT scheduling for identical processor systems. RAIRO Tech. Sci. Inform. 1, 69–75 (1982)
7. Liu, M., Chen, D., Wu, C., Li, H.: Fuzzy reasoning based on a new fuzzy rough set and its application to scheduling problems. Computers & Mathematics with Applications 51(9-10), 1507–1518 (2006)
8. Mllestad, T., Skowron, A.: A Rough Set Framework for Data Mining of Propositional Default Rules. In: Michalewicz, M., Raś, Z.W. (eds.) ISMIS 1996. LNCS, vol. 1079. Springer, Heidelberg (1996)
9. Piech, H., Ptak, A., Machura, M.: Evaluation of Operation and State of an Object Using Artificial Intelligence tools, pp. 465–475. Springer Science + Business Media, New York (2005)
10. Piech, H., Ptak, A., Machura, M.: Interchangeable Strategies (without Side Payments) on the base of Uncertain Information on Resources. Image Anal., Comp. Graph., and AI Applications, Białystok, 145–157 (2005)
11. Piech, H.: Scheduling with applying rough sets theory. Czestochowa University of Technology (2008)
12. Syslo, M.M., Deo, N., Kowalik, J.S.: Algorytmy optymalizacji dyskretnej. PWN, Warszawa (1995)

# The Problem of Bandwidth Allocation in the Business Activity of Service Providers: Comparison and Analysis of Costs

Bartosz Marcinkowski and Piotr Ostrowski

University of Gdansk, Department of Business Informatics, Piaskowa 9, 81-864 Sopot, Poland
bartosz.marcinkowski@univ.gda.pl
piotr.ostrowski@univ.gda.pl

**Abstract.** One of the fundamental questions in the business activity of service providers is how to allocate existing resources in a WAN computer network in order to meet customers' requirements and manage the bandwidth effectively. The aim of the current paper is to compare the effectiveness of software and hardware-based bandwidth allocation solutions in real-life situations. In Sect. 1, bandwidth allocation solutions selected for the test are introduced. Section 2 addresses costs that an internet provider had to sustain in order to implement individual solutions. Section 3 summarizes installation and configuration issues. In Sect. 4 performance levels achieved are assessed, while in Sect. 5 – additional capabilities and statistics are discussed. The article is concluded with a tabular comparison of discussed approaches.

## 1 Introduction

One of the fundamental questions in the activity of a service providers is how to allocate existing resources within a WAN computer network in order to meet customers' requirements and manage bandwidth effectively. The aim of the current paper is to compare the effectiveness of software and hardware-based bandwidth allocation solutions.

It was Trustix Secure Linux, along with bandwidth allocation scripts developed by Intrux, that was the software solution selected for the test. Hardware solutions were represented by the STM-1550 Standalone Traffic Manager from Archer's EtherWerX platform, a high-end carrier-class bandwidth allocation device. The scripts mentioned are intended to filter the traffic on the 7th layer of the ISO/OSI model. For a comparison, the software solution was implemented on two different machines:

– a HP ProLiant ML 150 G3 server with an Intel Xeon 5110 Dual Core 1.60 GHz processor, 1 GB of RAM memory, a 160 GB hard disk and two NICs, one built-in and the other the Intel FastEthernet 10/100S;
– s standard PC, acting as a server, with a 2 GHz AMD processor installed on an Asrock motherboard, 512 MB of RAM memory, a 40 GB hard disk and Realtek 8139-based NICs.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 318–324, 2009.

The STM-1550 Standalone Traffic Manager hardware-based bandwidth allocation solution enabled 2nd ISO/OSI layer traffic to be filtered. Such devices are suitable both for small (up to 100 hosts) and relatively large (more than 10,000 hosts) computer networks.

The survey conducted took into account not only technical effectiveness, but also such factors as quality-to-price ratio, reliability, scalability and flexibility of configuration. Survey infrastructure was based on a 10-Mbps symmetrical connection with over 300 hosts online. The scope of the test included monitoring the following:

– the processor overhead of all the devices taking part in the bandwidth-sharing activity;
– the effectiveness of bandwidth sharing and its optimal use;
– the amount of work necessary to modify and implement different network configurations;
– additional functionality, such as user-banning capability or http-based notifications on a selected host;
– statistics concerning the network connection and system usage level provided by both solutions.

## 2 Costs of Selected Solutions' Implementation

As mentioned before, the survey was conducted using a 10-Mbps symmetrical connection with more than 300 hosts online. Such scale of implementation may be classified as typical when small enterprises – offering services on bounded geographical areas (such as housing estates) – are in discussion. For a small enterprise, and its not particularly impressive budget, cost of initial investment in a selected bandwidth allocation solution has a critical meaning.

As a rule, software solutions are cheaper than hardware ones. Such factors as server cost, operating system cost and bandwidth allocation scripts have to be taken into account in a calculation. It is installing an open source operating system that is among the most common practices. Variety of Linux distributions provide full support for software solutions.

First of options compared represented a software solution based on a PC-class computer with 2 GHz AMD processor that was available for not more than 1,500 PLN. Secure Trustix Linux distribution installed is based on a GPL license, and therefore it is free of charge. Professional bandwidth allocation scripts prepared by Intrux cost only 50 PLN, what makes a total cost of 1,550 PLN more than attractive from a small company's point of view.

The second software option, incorporating a typical HP ProLiant ML 150 G3 server with an Intel Xeon 5110 Dual Core 1.6 GHz processor, is a more reliable but also significantly more expensive one. Obtaining such a machine required a budget of 4,100 PLN while carrying out the survey. Along with the Intrux scripts budget summarizes with 4,150 PLN.

Hardware solution incorporates a professional Standalone Traffic Manager device (STM-1550 model to be exact) along with a dedicated EwxOS® operating system. STM-1550 is a member of EtherWerX platform developed by Archer. The net value of the device itself reached 1,649 € (about 7,850 PLN with tax included) while conducting the survey. User is allowed to connect up to 100 hosts without a charge. For each 100 additional hosts in a LAN that is up-linked to the Internet, a license is required. Such a license costs 299 € (about 1,420 PLN with tax included). Assuming that the scale of a computer network managed by a typical company exceeds 300 users, total cost of the hardware solution along with licenses is no less than 12,110 PLN. Moreover, the cost of a network server that is supposed to host such services as NAT/PAT, DHCP, WWW and DNS ought to be included. Since listed services do not require sophisticated hardware, keeping in mind the scale of the enterprise, low-end machine may be purchased (about 1,000 PLN). Budget should therefore be increased to 13,110 PLN.

Summing up the costs of selected solutions' implementation, software solutions are indeed more attractive in this aspect. This fact itself practically disqualifies hardware solutions in the initial stages of company's development. Difference in price per user is not significant enough though to compensate contingent relevant weaknesses that may be revealed while testing other aspects of the presented solutions.

## 3 Installation and Configuration

The process of operating system installation along with implementation of bandwidth allocation scripts is the same for both software solutions. The activity occupies system administrator for over an hour, given that the administrator is at least moderately proficient in Linux installation and management. While conducting the test, authors did not come across any issues within installation process – in either of the cases. Thanks to the Intrux manual (PDF format, 11 pages) explaining bandwidth allocation scripts as well as other supported services usage, both initializing network and implementing configuration changes are relatively easy. In each of the files provided, every configuration parameter is supplemented with a detailed synopsis, what reduces the duration of configuration activity. Total implementation time, aside from equipment connection, sums up to about 2 hours.

When hardware solution is in discussion, STM-1550 device is shipped with EwxOS® operating system pre-installed. The system is dedicated to support exclusively the device, and – if buyer wishes so – may be properly configured by the vendor. Thanks to that, the equipment is ready to use just after finishing physical connection.

STM-1550 parameters may be intuitively modified, but in order to achieve that, the system administrator must become familiar with extensive manual, explaining possible values of individual properties. As more than 220 pages are contained in the manual, significant amount of time should be reserved.

## 4 Performance Levels

It is performance that is one of the key factors in the selection of optimal bandwidth allocation solution for providing Internet access in a computer network. In most cases, analysis of overall performance is followed by preparing a performance-to-price ratio and using it for comparison.

In the analysis performed, the first option (which was a PC-class computer with 2 GHz AMD processor and 512 MB of Random Access Memory) proved to be highly unsatisfactory when performance is in discussion. Slightly more than 300 connected hosts caused a processor overhead between 85% and 100% in the peek hours. Consequently, some system processes were disabled from time to time and provided services were out of order and back again. Average transfer rates dropped as well. After reducing the number of connected hosts to 200, processor overhead did not exceed 50% and the system became stable. All provided services were online all the time. Said so, implementing this option in more complex networks is definitely not recommended.

HP server as a subject of testing achieved much better results. With all 300+ hosts online, the processor overhead remained between 15% and 20%. Naturally, such a level caused no side effects. The system was perfectly stable and had a lot of computing power in reserve for additional services and/or hosts. Described solution was adequate for the assumed network scale. When only 200 hosts were using the services provided by the server, monitored processor overhead did not exceed 10%, being close to 5% most of the time.

Hardware solution, i.e. STM-1550, used between 20% and 25% of its computing power to manage a 300+ hosts network. Although the result is similar to the HP server, hardware device was designed to manage bandwidth allocation only, so additional required services had to be transferred to the other machine (PC or server). In the test performed, the same AMD computer, as in the first case, was used. As a result of transferring bandwidth management to a dedicated STM-1550 device, PC's processor overhead was about 30%. Said so, hardware solution is highly scalable, allowing a radical increase in number of host connected to the Internet.

The quality of the data transfer and bandwidth allocation efficiency was generally comparable between all solutions. It should be noted though that the processor overhead of the PC that was used for testing the first option caused a random interruptions in service operation. Despite the fact that the very idea of software solutions (traffic filtering based on 7th ISO/OSI layer information) is definitely different from hardware solutions (based on 2nd layer), individual users were unable to tell which solution was online at the moment. That is because individual connections' properties were equally good no matter which of the solutions was implemented at the moment. Bandwidth is allocated fair and square depending on the current Internet uplink load. Each host had a maximum download and upload speed set to 2 Mbps, while minimal guaranteed speed was not configured (this option was set to "auto", what results in smooth adjustment of the speeds of individual hosts links to the current traffic in the network. Host download speed was measured by downloading a 100 MB file from the chosen testing website, http://noc.gts.pl. In the peek hours, both hardware solution and software solution based on the HP server acquired a download speed of 600–800 kbps. Relatively high download speeds result from the diversity of used network services (WWW, e-mailing, VoIP, P2P). Not all of the mentioned services use connection in the continuous manner, WWW for instance generates traffic mainly while opening a new website. Speeds measured for standard PC-based software solution, with 85%–100% processor overhead, varied drastically, ranking 200–400 kbps average.

## 5 Additional Capabilities and Statistics

In the test performed, additional capabilities and statistics offered both by scripts and STM-1550 operating system were taken into account for comparison. Service provider's everyday practice indicated that the administrator is frequently forced to use a certain set of options and tools. The hybrid of Trustix Secure Linux operating system and bandwidth allocation scripts was found to be the most flexible basis for administrative tasks. Such set allows performing a wide range of activities. Among them were:

– banning individual users with immediate display of the configured message on their machines (lack of payment for instance);
– periodical display of the chosen message on the users' Internet browsers;
– mapping public IP address with one or more private IP addresses;
– blocking certain ports.

Mentioned set includes not only bandwidth allocation service, but also a firewall that protects the local area network. Such functionality not necessarily could be expected in a bandwidth manager software, but it makes a very fine addition. Broad functionality of a tool set is what matters for a small enterprise, and a software firewall would have to be obtained and implemented on the server anyway.

All kinds of statistical tools had to be installed additionally in TSL operating system. Therefore the statistics are not discussed and assessed, because such functionality is fully dependant on software provider. EtherWerX STM-1550 bandwidth manager introduces basic statistical capabilities, encompassing current total transfer rate as well as upload/download per host. EwxOS® provides statistics concerning each user and system load via SNMP protocol. Thanks to data obtained via SNMP, graphical charts may be generated. Exemplary charts include:

– each link load,
– each channel load,
– temporary system load caused by each user,
– number of active users in a given time frame,
– processor overhead,
– system resources usage,
– temporary load resulting from the number of transmitted packets.

On the other hand, acquiring SNMP protocol data requires obtaining adequate MIB database from the vendor as well as performing advanced configuration activities.

## 6 Summary

After performing analysis of all the test components, authors suggest considering high-end software solutions while selecting bandwidth allocation mechanisms. Such solutions are more flexible, easier to support, require smaller budget and finally – are not necessarily less effective than hardware solutions based on EtherWerX STM-1550 bandwidth manager. It is finding the appropriate server properties that is one of the key activities in the initial implementation stage. One should also take into account solution scalability – slight upsizing is justified. Brief comparison of discussed solutions is summed up in Table 1.

**Table 1.** Comparison of selected software and hardware bandwidth allocation solutions

| | Standard PC with AMD 2 GHz processor + Intrux scripts | HP ProLiant ML 150 G3 + Intrux scripts | STM-1550 hardware solution |
|---|---|---|---|
| Cost [PLN] | 1,550 | 4,150 | 13,110 |
| Difficulty level of configuration implementation | moderately advanced | moderately advanced | easy |
| Basic configuration time | about 2 hours | about 2 hours | none (implemented by the vendor) |
| Processor overhead in peak hours | 85%–100% | 15%–20% | 20%–25% |
| System stability | very low | high | high |
| Solution functionality | high | high | moderate |
| Subjective weighted rating | 1 | 10 | 6 |

Along with computer network growth, migration to hardware solutions may become inevitable. It is reliability of hardware bandwidth allocation solutions that is their greatest advantage. In practice, a LAN network ought to radically exceed the scale of 300 hosts to make such migration well-founded (taking costs into account).

## References

1. Dye, M., McDonald, R., Rufi, A.: CCNA Exploration Network Fundamentals, 2nd edn. Cisco Press (2007)
2. Graziani, R., Johnson, A.: Routing Protocols and Concepts, CCNA Exploration Companion Guide. Cisco Press (2007)
3. http://noc.gts.pl
4. Marcinkowski, B., Ostrowski, P.: Current Tendencies in Providing Broadband Internet Access based on FastWan Company (in Polish). Prace i Materialy Wydzialu Zarzadzania UG, Gdansk (2007)
5. Marcinkowski, B., Ostrowski, P.: IP-Based Television – Development and Recognition Level in Poland. Technologie Informacyjne 14, Gdansk (2007)
6. Ostrowski, P., Marcinkowski, B.: Evolution of Professional Work-Related Priorities of Pomeranian Regional Cisco Academy Students. Company in International Environment, Gdansk (2008) (in Polish)
7. Winiarski, J. (ed.): Computer Networks in Business (in Polish). Gdansk University Press, Gdansk (2008)

# Network Transmission of 3D Mesh Data Using Progressive Representation*

Krzysztof Skabek and Łukasz Ząbik

Institute of Theoretical and Applied Informatics
Polish Academy of Sciences,
ul. Bałtycka 5, 44-100 Gliwice, Poland
kskabek@iitis.pl, lukasz.zabik@gmail.com
http://www.iitis.pl

**Abstract.** In this chapter we focused on *progressive meshes* which are the way to hierarchically represent 3D objects. We present the *Me3d* system developed to process and present digitalized 3D objects. The distributed transmission of 3D meshes over the communication line is discussed.

## 1 Introduction

There are some ways to make the processing and transfer of complex 3D objects more efficient. In one of the approaches we use hierarchical representation of object with several levels of details defined. First, only the basic data is transmitted just to see the shape of the object. The more detailed model can be transfered on demand, e.g. when viewer is coming closer. Transmitting a further detailed data of the mesh over communication line one may obtain the model at the higher level-of-details [6].

There are many different ways to represent graphical 3D models. In this chapter we focused on *progressive meshes* which were introduced by Hoppe [3]. We assume here that the progressive mesh complexity does not depend on viewer position, it is applied to the whole surface.

In fact meshes are only the polygonal approximations of physical objects, but thanks to progressive representation they can be stored and transfered with fluently changing accuracy. Detailed meshes are often obtained by scanning physical objects using range scanning systems. In this case, the resulting complex meshes are expensive in storing, transmitting and rendering. Progressive representation solves many practical problems. First of all it simplifies the mesh structure (mesh simplification). Another improvement is defining approximations for certain levels-of-details (LOD), so that it is easier to display the more detailed model when the observation point is getting closer. During transmission over a communication line the user can choose certain level-of-details and after that he can decide to enhance the structure transmitting only the additional mesh

* This work was supported by the Ministry of Science and Higher Education of Polish Government, the research project no N N516 1862 33.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 325–333, 2009.

data. And finally, as the mesh storing is very memory consuming, the progressive representation may be used to compress meshes.

In the chapter the Me3d system developed in our Institute was described. The program is used to process and present digitalized objects of cultural heritage. A module for building and presenting the progressive meshes is its component. Storing data in the progressive meshes is a way to hierarchically represent 3D objects. Such approach is being developed and extended in the research project N N516 1862 33[1] and its aim is to store and virtually distribute objects of cultural heritage.

## 2 Mesh Representation

A mesh is the basic representation of 3 dimensional graphical objects [7]. We focused on triangle meshes where each face is build of 3 vertices. The basis of the representation is a set of point coordinates $(x, y, z)$. The orientation of faces as well as storing order of vertices is very important. Therefore the normal vectors for faces and the ordered vertices are also used to define the mesh properties.

The mesh geometry can be denoted by a tuple $(K, V)$ [2], where $K$ is *a simplicial complex* specifying the connectivity of the mesh simplices (the adjacency of the vertices, edges, and faces), and $V = \{v_1, \ldots, v_m\}$ is the set of vertex positions defining the shape of the mesh in $R^3$. More precisely, we construct a parametric domain $|K| \subset R^m$ by identifying each vertex of $K$ with a canonical basis vector of $R^m$, and define the mesh as the image $\phi_v(|K|)$ where $\phi_v : R^m \rightarrow R^3$ is a linear map.

Besides the geometric positions and topology of its vertices, the mesh structure has another appearance attributes used to render its surface. These attributes can be associated with faces of the mesh. A common attribute of this type is the material identifier which determines the shader function used in rendering a face of the mesh. Many attributes are often associated with a mesh, including diffuse colour $(r, g, b)$, normal $(n_{\mathrm{x}}, n_{\mathrm{y}}, n_{\mathrm{z}})$ and texture coordinates $(u, v)$. These attributes specify the local parameters of shader functions defined on the mesh faces. They are associated with vertices of the mesh.

We can further express a mesh as a tuple $M = (K, V, D, S)$, where $V$ specifies its geometry, $D$ is the set of discrete attributes $d_f$ associated with the faces $f = \{j, k, l\} \in K$, and $S$ is the set of scalar attributes $s(v, f)$ associated with the corners $(v, f)$ of $K$.

As many vertices may be connected in one corner with the same attributes, the intermediate representation called *wedge* was introduced to save the memory [4]. Each vertex of the mesh is partitioned into a set of one or more wedges, and each wedge contains one or more face corners. Finally we can define the mesh structure that contains an array of vertices, an array of wedges, and an array of faces, where faces refer to wedges, and wedges refer to vertices. Face contains indices to vertices, additionally this structure contains array of face neighbours (*fnei*) in which indices of tree adjacent faces are stored; this information is necessary to

[1] http://www.iitis.pl/zksw/vm_project/

build a progressive mesh. There is nothing said in original papers about order of vertices and indexes of adjacent faces in Face structure. In our implementation the counter is stored clockwise and additionally first adjacent face is at first position as first vertex, so if we cross first edge we find the first neighbour, if we cross second we find the second, etc.

In many places of this chapter we use the word *edges*. The edge is a connected pair of vertices or, in other words, it is a pair of adjacent vertices. There is no additional list of edges, but the first vertex and the face to which this edge belongs are defined instead. Using *wedge* we can access vertex, even if the adjacent face does not exist we can define edge. Definition of edges is necessary to simplify meshes, to create progressive meshes as well as to determine which edge (or vertex) could be collapsed.

### 2.1 Manifold Surfaces

Surfaces are often, as well as in our approach, assumed to be manifolds. A manifold [7] is a surface, all of whose points have a neighbourhood which is topologically equivalent to a disk. A manifold with boundary is a surface all of whose points have a neighbourhood which is topologically equivalent to either a disk or a half-disk. A polygonal surface is a manifold (with boundary) if every edge has exactly two incident faces (except edges on the boundary which must have exactly one), and the neighbourhood of every vertex consists of a closed loop of faces (or a single fan of faces on the boundary).

Many surfaces encountered in practice tend to be manifolds, and many surface-based algorithms require manifold input. It is possible to apply such algorithms to non-manifold surfaces by cutting the surface into manifold components and subsequently stitching them back together. However, it can be advantageous for simplification algorithms to explicitly allow non-manifold surfaces.

### 2.2 Progressive Mesh

*Progressive mesh* (PM) [3] is special case of the mesh or rather extension of the mesh, it makes it possible to build mesh of different level-of-details (LOD) [6]. It also allows us loading base mesh $M^0$, the mesh of lower LOD (explained later in more details), and then progress loading of the remaining parts of the mesh. As an input source of loading we may use memory input stream.

In PM form, an arbitrary mesh $\widehat{M}$ is stored as a much coarser mesh $M^0$ together with a sequence of $n$ detail records that indicate how to incrementally refine $M^0$ exactly back into the original mesh $\widehat{M} = M^n$. Each of these records stores the information about *a vertex split*, an elementary mesh transformation that adds an additional vertex to the mesh. Thus the PM representation of $\widehat{M}$ defines a continuous sequence of meshes $M^0, M^1, \ldots M^n$ of increasing accuracy, from which LOD approximations of any desired complexity can be efficiently retrieved. Moreover, smooth visual transitions (*geomorphs*) [3] can be efficiently constructed between any two such meshes. In short, progressive meshes offer an efficient, lossless, continuous-resolution representation. Progressive meshes makes it possible not only to store the geometry of the mesh surface, but, what is more important, preserve its overall appearance, as defined by the discrete and scalar attributes associated with the surface.

There are three operations that make it possible to determine the base mesh of $\widehat{M}$: edge collapse, vertex split and edge swap. Edge collapse operation is sufficient to successfully simplified meshes. An edge collapse operation $ecol(v_s, v_t)$ remove one edge and instead two vertices $v_s$ and $v_t$ insert new one $v_s$ (see Fig. 1). Additionally two faces $(v_t, v_s, v_l)$ and $(v_t, v_r, v_s)$ are removed. The initial mesh $M_0$ can be obtained by applying a sequence of n edge collapse operations to $\widehat{M} = M^n$:

$$(\widehat{M} = M^n) \overset{ecol_{n-1}}{\rightarrow} \cdots \overset{ecol_1}{\rightarrow} M^1 \overset{ecol_0}{\rightarrow} M^0$$

The edge collapse operation is invertible. The inverse transformation is called vertex split. Vertex split operation adds in place of vertex $v_s$ two new vertices $v_s$ and $v_t$ and two new faces $(v_t, v_s, v_l)$, $(v_t, v_r, v_s)$ if edge $\{v_s, v_t\}$ is boundary then adds only one face (see Fig. 1). Because edge collapse transformation is invertible our mesh $\widehat{M}$ can be presented as a simple $M^0$ and sequence of $n$ vsplits records:

$$M^0 \overset{vsplit_0}{\rightarrow} M^1 \overset{vsplit_1}{\rightarrow} \cdots \overset{vsplit_{n-1}}{\rightarrow} (\widehat{M} = M^n)$$

We call $(M^0, vsplit_0, ..., vsplit_{n-1})$ a progressive mesh (PM) representation of $M$.

Further description of progressive mesh representation and its methods can be found in [2,3,4]. Our implementation of these structures is described in [5].

## 3 Strategy of Network Transmission

The mesh data for progressive representation is stored as a special structure `Vsplit`. It stores the transition data for operations *Vertex Split* and *Edge Collapse*. The class is based on the structure proposed by Hoppe [3] and was extended by certain elements. Particularly the fields of `Vsplit` structure are described in Fig. 1.

In the structure faces are described by its identifiers, e.g. `f11_id` (see Fig. 1). Variables `f1_vt_i` and `f2_vt_i` concern faces `f1_id` and `f2_id` accordingly. These are indexes of vertex $v_t$ in vertex table for faces $f_1$ or $f_2$, they can assume values 0, 1 or 2.

A single `Vsplit` instance takes 80 Bytes. The transmission of `Vsplit` packets is proportional to the number of vertices incoming to the structure and must be followed by some additional information such as number of objects, which only slightly increase the volume.

The initial transmission of the progressive mesh is arranger as a sequence of the base mesh $M^0$ and the *VSplit* records up to the assumed level-of-detail. Another transmissions include only the *VSplit* records.

```
class Vsplit {
    public:
      s32 flclw;
      s8 vlr_rot;
      s32 f11_id;
      s32 f21_id;
      s32 f22_id;
      s8 f1_vt_i;
      s8 f2_vt_i;
      struct {
        s8 vs_i;
        s8 corners;
        s8 ii;
        s8 matid_predict;
      }code;
      s8 fl_matid;
      VertexAttribD vad_l, vad_s;
      E3DVectorWedgeAttribD wads;
  };
```

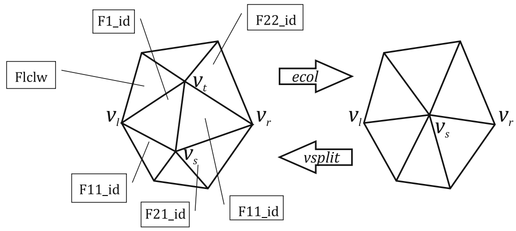



**Fig. 1.** `Vsplit` structure

## 4 Me3d Framework

Me3D application was designed to process the mesh structures. The application is built in Visual Studio style. It consists of main window and many dockable windows – panels (see Fig. 2).

Using the application it is possible to view and process 3D objects stored in several file formats: *stl*, *obj*, *ply*, *vrml*. Native *xml*-like formats (*me3d* and *me3db*) are also defined to store the mesh and point data (also for progressive representation). Separate 2D views can be also caught and saved to bitmap files. The opened objects are listed in *Project panel*. It is also possible to store the appropriate scene settings (rotation, shift, zoom). There are several operations that can be performed on object. First of all we can pick the object and then interactively select its parts (faces, vertices, etc.) using *Picking panel*. We can stretch object in every direction (axis x,y,z), move, and rotate around arbitrary axis (x,y,z) using *Apply Transform panel*. The most interesting here is rotating the object around any arbitrary line using *Rotate About Given Line panel*. The program makes it possible to display objects in three modes, as: vertices, edges or triangular faces. Program can work in two modes: *Normal* or *Edit* mode. In the first mode we can view object form any direction, in the second one we are able only to zoom object and select its parts. *UnDo* and *ReDo* operations are available in *Edit* menu. Next we can also set the visibility of the additional useful objects: axes, boundary boxes and the grid. It is also possible to reset the view.

In *Project panel* we have the list of open meshes. Each mesh can be visible and/or active. If the mesh becomes invisible it also becomes inactive. But the meshes can be also inactive and visible.

*Object properties panel* informs us whether the object is visible or not, and the same with active property. The number of faces and vertices in the mesh is displayed. We can find also the size of the object. Below the properties window there is a view manager where we can save same interesting views and retrieve them later.

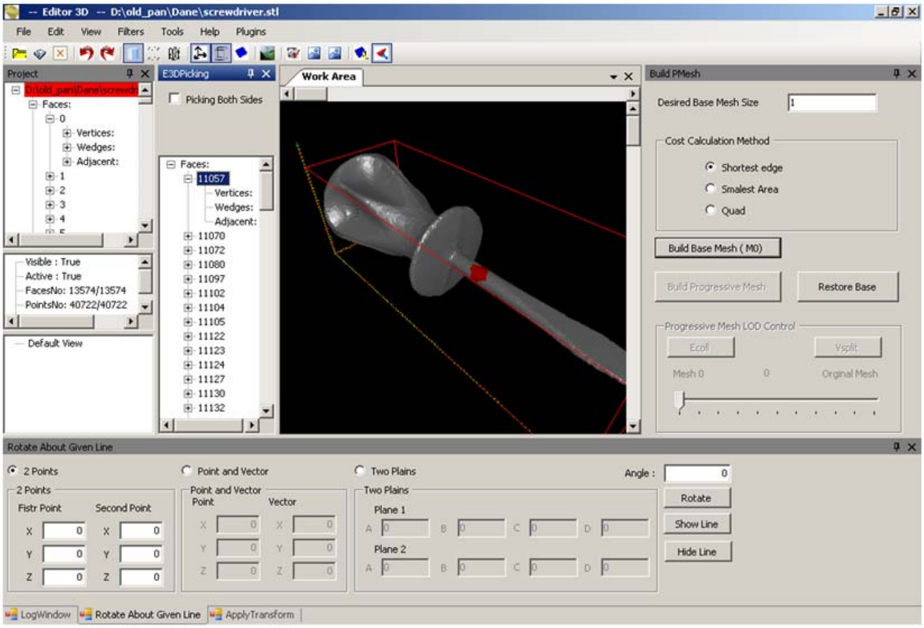



**Fig. 2.** Me3d Framework

There are two panels defining basic object transformations: translation, rotation, scale and also rotation around the given axis. There are three methods of the rotation around line, they require alternatively : (1) setting two points lying on the line, (2) setting one point and a vector, (3) setting two planes that are not parallel and the line of their intersection. Of course in each case we give the angle of rotation in degrees.

The progressive mesh can be obtained and processed using *Build PMesh panel*. There are some steps to be fulfilled. First of all we define the size of the desired base mesh $M_0$ using edit box *Desired Base Mesh Size*. We choose the method of cost function calculation. There are three methods implemented: (1) the shortest edge, (2) the smallest area and (3) the quadric distance [1] (which gives the best results). After that the dialog group *Progressive Mesh LOD Control* will become active and we can manipulate the complexity of the resultant progressive mesh. It can be done in two ways: using buttons *Ecoll* and *Vsplit* we change the structure by one step or using track bar.

When we enter *Select Faces* mode, we can see the selection in *Picking panel*. There is only one box there: *Picking Both Sides*. Using this box we set if we want to select all faces in the selected area or only the visible faces. In the tree of the panel we see identifiers of selected faces. We can also see the selected faces marked in different colour in the working area.

The latest update for the framework makes it possible to display textures and convert them into face colors of the mesh.

## 5 Architecture of Me3d Framework

The architecture of Me3d system is shown in Fig. 3. The system has a modular structure. Three modules can be distinguished: (1) mesh container, (2) 3D displaying module and (3) GUI. The first module is responsible for object representation, processing and storing. The second module uses OpenGL (TAO) and Glut libraries to draw the 3D object in the screen. The third module performs operations on the interface. The system makes it possible to design user plug-ins in form of *dll* modules. There is a *Plug-in manager* included to dynamically load/unload software modules.

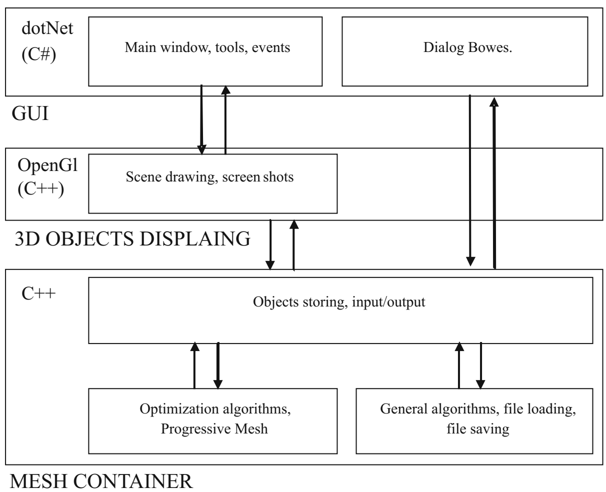



**Fig. 3.** Me3d system architecture

The software extensions and libraries used in the project are as follow: (1) *Microsoft Visual Studio 2008* – the development environment, (2) *.NET Framework*, (3) *Tao* – OpenGL control library for .NET, (4) *Glut* – fast and portable graphical library and (5) *Pthread* – synchronization library.

## 6 Tests and Results

Experiments were performed on 3D meshes of digitalized physical objects. In Fig. 4 we presented the simplification of bowl from archaeological collection of Museum in Gliwice, Fig. 5 presents the simplification of figure named *Sabines* from Art Department of the Museum.

Tests were performed using processor Intel T2130 1.8 GHz, 2 GB RAM and NVidia graphic card with 256 MB RAM. During tests the volume of packages at different levels-of-details was checked. The results for the tested meshes are shown in Table 1.

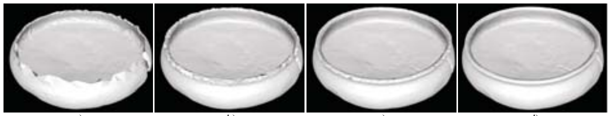



**Fig. 4.** Object – archeological bowl from Museum in Gliwice

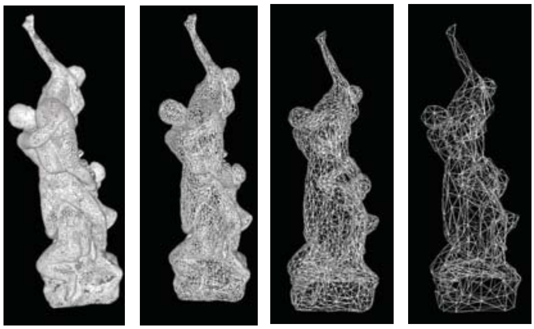



**Fig. 5.** Object – figure of Sabines from Museum in Gliwice

**Table 1.** Results for the tested meshes

| Number of faces | Base mesh size [kB] | Difference size [kB] | Pack size [kB] | Vertex split size [kB] |
|---|---|---|---|---|
| Bowl | | | | |
| 41 400 | 10 512 | 836 | 836 | 0 |
| 20 000 | 5 078 | 391 | 1 227 | 836 |
| 10 000 | 2 539 | 195 | 1 422 | 1 227 |
| 5 000 | 1 270 | 98 | 1 520 | 1 422 |
| 2 500 | 635 | 49 | 1 568 | 1 520 |
| 1 250 | 269 | 16 | 1 585 | 1 568 |
| 828 | 178 | 178 | 1 763 | 1 585 |
| Sabines | | | | |
| 107 682 | 27 341 | 2 253 | 2 253 | 0 |
| 50 000 | 12 695 | 977 | 3 230 | 2 253 |
| 25 000 | 6 347 | 488 | 3 718 | 3 230 |
| 12 500 | 3 173 | 244 | 3 962 | 3 718 |
| 6 250 | 1 586 | 186 | 4 148 | 3 962 |
| 1 500 | 380 | 380 | 4 528 | 4 148 |

In Fig. 6a the volume of mesh package for different levels-of details (LOD) is presented. It was assumed that the base mesh is a first object we download, a package is a list of vsplits necessary to reconstruct the original object and difference package is a set of vsplits necessary to reconstruct structure at the next LOD.

Using our application we obtained a collection of objects with instances of different LOD. Assuming the fixed speed of network connection we can calculate

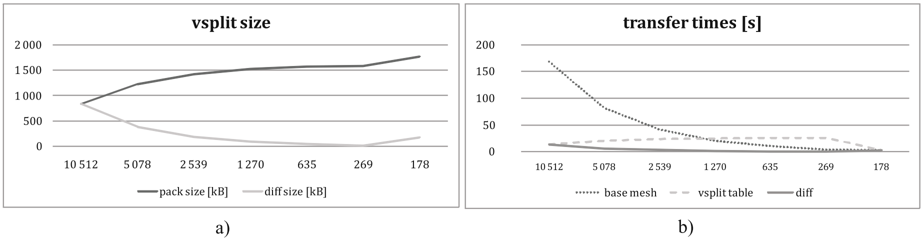



**Fig. 6.** Comparison of mesh packages at different LODs: (a) volume, (b) transfer time

transfer times of the meshes at different levels-of-details. The results for transfer speed 512 kb/s are presented in Fig. 6b. Additionally, time necessary to transfer the structure at the given LOD without prior simplification is shown.
Results of our tests lead to the following conclusions:

- Progressive meshes can significantly speed up loading 3D graphic objects via network connection, viewer is able to watch object as soon as the base mesh is loaded, the object may be of very pure quality but will be increased progressively.
- Bigger package granulation allow us to observe the object rebuilding in real time, i.e. computer software downloading 3D graphic object performs a set of *VSplit* operations and reconstruct object, if, on the other hand, smaller packages are transferred the object reconstruction is more fluent.
- Progressive mesh is a structure that compresses graphical 3D objects, we can observe it comparing base mesh size with package necessary to rebuild mesh.

Techniques described in the chapter may be used to present large graphical object on websites, it is a very efficient way of transferring object over network.

## References

1. Garland, M.: Quadric-Based Polygonal Surface Simpling. School of Computer Science Carnegie Mellon University, Pittsburgh (1999)
2. Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., Stuetzle, W.: Mesh Optimization. In: Computer Graphics, SIGGRAPH 1993 Proceedings, pp. 19–26 (1993)
3. Hoppe, H.: Progressive Meshes. In: Computer Graphics, SIGGRAPH 1996 Proceedings, pp. 99–108 (1996)
4. Hoppe, H.: Efficient Implementation of Progressive Meshes. Computer & Graphics 22(1), 27–36 (1998)
5. Skabek, K., Ząbik, Ł.: Implementation of Progressive Meshes for Hierarchical Representation of Cultural Artifacts. In: ICCVG 2008 Proceedings, Warsaw (2008)
6. Luebke, D., Reddy, M., Cohen, J.D., Varshney, A., Watson, B., Huebner, R.: Level of Details for 3D Graphics. Morgan Kaufmann, San Francisco (2003)
7. Nielsen, F.: Visual Computing: Geometry, Graphics and Vision. Charles River Media (2005)

# Application of Distributed System in Control and Diagnostic Toothed Gears

Andrzej Kwiecień[1], Jacek Rysiński[2], and Marcin Sidzina[2]

[1] Silesian University of Technology, Institute of Informatics
andrzej.kwiecen@polsl.pl
[2] University of Bielsko-Biala,Department of Mechanical Engineering Fundamentals
{jrysinski,msidzina}@ath.bielsko.pl

**Abstract.** This chapter contains an analysis of a system for gear control and diagnosis using Ethernet-based distributed control system. Own-designed and made test stand was presented along with a concept of the device protection system operation.

## 1 Introduction

In order to ensure continuous operation of the machine it is necessary to monitor its condition. Diagnostic machine models show a common feature of interdependence between the state parameters and diagnostic signal. Irrespective of the diagnostic model assumed it is necessary to know the relations between the object state parameters and the diagnostic parameters. The relations are usually determined experimentally by analysing the operation history or carrying out diagnostic experiments. An attempt was made to construct a diagnostic station to monitor the condition of the machine and secure the system in case of emergency or failure. Whatever the system to be analysed, there are many symptoms which provide information on the condition of the mating components. They include noise, vibrations, temperature, change of load, change of ampere-hour efficiency etc. [3, 4, 5]. The concept of the station constructed consisted in monitoring of control signals and diagnostic signals (Fig. 1).

The major purpose was to develop a distributed diagnostic system to allow evaluation of the machinery condition using wide area networks. Control and signal acquisition algorithms were implemented in a PLC of GE Fanuc. Data archiving and detailed analysis of symptoms was performed using a PC-class computer with the use of Ethernet. The solution proposed allows to collect the object state vector on the controller in real time and send it for further analysis by the PC. The designed system is characterized by multitasking as it enables simultaneous analysis of signals supplied by a number of devices. While the evaluation of a change in temperature, rotational speed and force is possible on the controller, the analysis of vibrations or noise is impossible to be carried out in a simple manner as numerical calculations must be performed which require higher processing capacities. Furthermore, the knowledge base related to processes occurring during the operation of the machine will e.g. allow to

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 334–343, 2009.

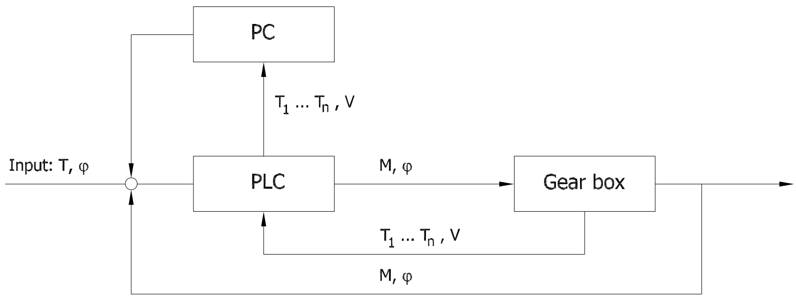



**Fig. 1.** Block scheme: $T$ – temperature [K], $M$ – moment [Nm], $\varphi$ – rotational speed [$s^{-1}$], $V$ – vibration

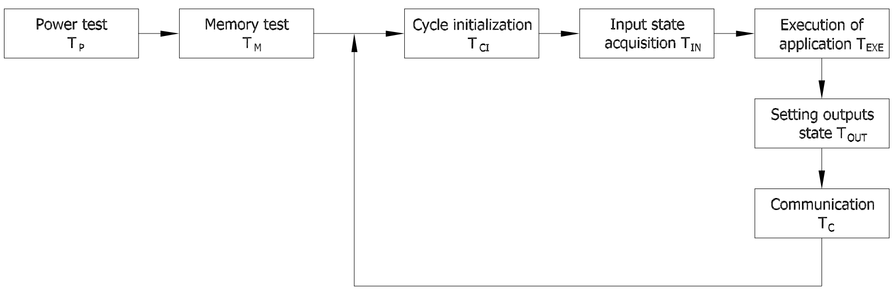



**Fig. 2.** Operating cycle of a freely programmable controller

schedule periodical maintenance. Experimental verification was conducted on a cylindrical gear.

The freely programmable PLC controllers, commonly used in control systems, have to collect signals from an object to obtain the current state of the process, necessary for the control system [1, 2]. Therefore, it is more and more frequent that the signals collected by the PLCs are used for diagnostics of the performing system. Modern, freely programmable controllers have larger and larger processing powers, enable real-time recording and analysis of numerous symptoms occurring during the operation of a machine. Originally PLCs were intended to replace incorporated logic circuits. Currently, those device are capable of processing analogue signals and have many communication possibilities.

In the PLCs the acquisition of variable states from the object and setting of output states for the performing devices comes down to tasks performed in cycles. Set of all tasks performed was called a cycle of the freely programmable controller (Fig. 2). Each task was assigned execution time which impacts the duration of the controller cycle loop.

$$T_{\mathrm{A}} = T_{\mathrm{CI}} + T_{\mathrm{IN}} + T_{\mathrm{EXE}} + T_{\mathrm{OUT}} + T_{\mathrm{C}} \qquad (1)$$

where:

$T_{\mathrm{A}}$ – duration of a freely programmable controller's cycle;
$T_{\mathrm{CI}}$ – duration of additional operations such as controller diagnostics;
$T_{\mathrm{IN}}$ – time designated for serving of controller inputs;
$T_{\mathrm{EXE}}$ – time designated for the execution of user's application;
$T_{\mathrm{OUT}}$ – time designated for serving of controller outputs;
$T_{\mathrm{C}}$ – time designated for communication.

The above is, so to say, a basis for conduct when selecting a control system for various machinery and equipment.

## 2 Test Stand

A test stand together with control and adjustment system was constructed in the Laboratory of Mechanical Engineering Fundamentals (of the Department of Mechanical Engineering Fundamentals of the University of Bielsko-Biala). The stand allows testing of phenomena occurring during the operation of an motor-gear system (Fig. 3, 4). An attempt was made to construct a diagnostic stand based on standard systems used in industrial practice. Parameters measured in the adjustment system included: torque on the gear input, vibrations of the case and temperatures at the characteristic points of the gear.

The system comprises: three-phase motor (1) of the power of 5.5 kW powered via frequency converter (5) including integrated overload-protection system. It is connected, via coupling (4) with the gear being tested (2) and a current generator on the output (3), via which the load is set. The adjustment system was

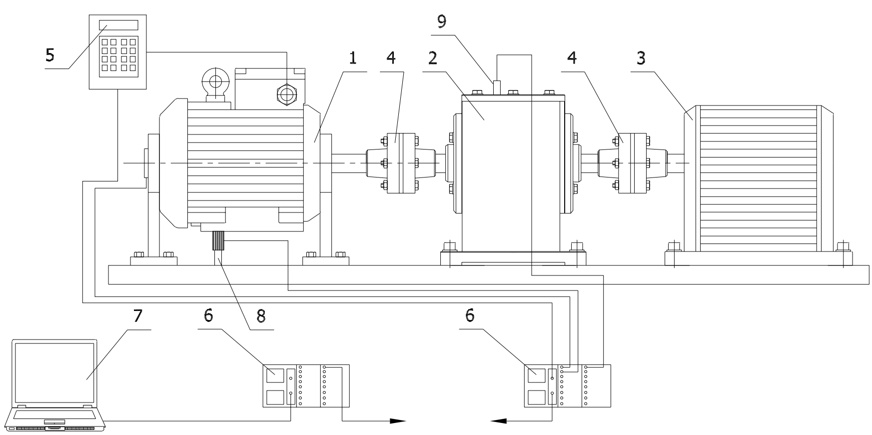



**Fig. 3.** Diagram of the measuring stand: 1 – three-phase motor, 2 – gear tested, 3 – DC motor, 4 – coupling, 5 – inverter, 6 – controller, 7 – data logger, 8 – force sensor, 9 – vibration sensor

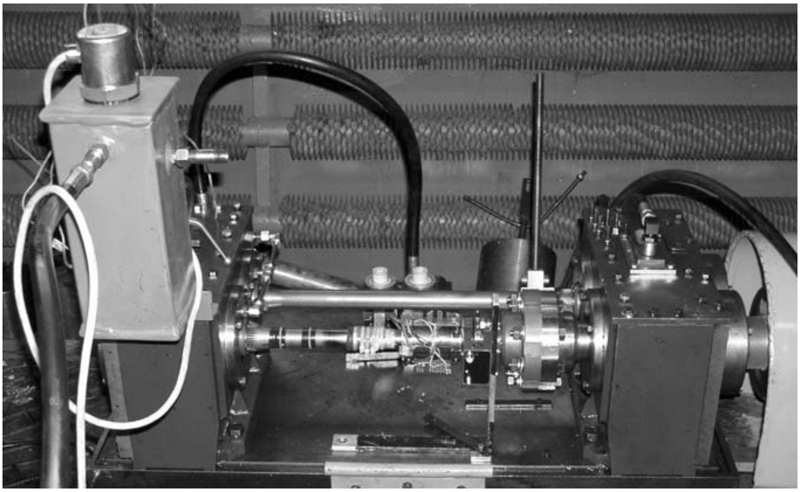
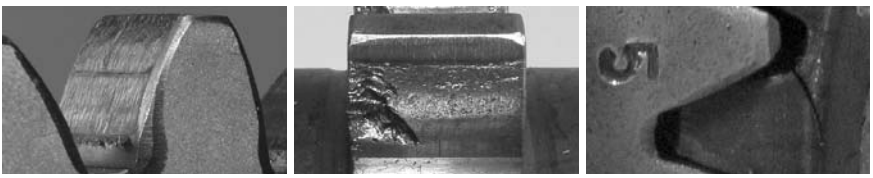


**Fig. 4.** Real test stand



(a) (b) (c)

**Fig. 5.** Defects in gears: (a) scoring, (b) breaking of gear, (c) pitting

constructed on the basis of a freely programmable controller of GE-Fanuc offering the possibility to transmit data over Ethernet according to MODBUS/TCP or SRTP transmission protocols. It allows continuous adjustment of rotational speed, turning the stand on/off, monitoring of damage symptoms and putting the device in a safe state in case of emergency.

The basic gear damage symptoms include temperature and vibrations [6, 7]. Operation of the gear results in degradation of side faces of meshing parts such as gear wheels, bearings etc. which shows itself in vibrations of the case walls and in the form of noise. For example noise caused by bearings is hardly audible. However, as the wear of the rolling parts, e.g. raceway, increases the noise level increases drastically to reduce the comfort of operation. It is, of course, the first symptom of damage which disables the gear from further operation. The basic reason for damage of wheels is the manufacturing technology and mechanical working. Noise emitted during the operation of wheels can be perceived as wailing, blasting or rattling.

In case of a worn gear new vibration causes occur such as: pitting, crumbling or breaking of teeth, bearing system errors or bearing damage etc (Fig. 5). In order to relatively quickly detect a fault it is necessary to have a pattern of vibrations on the gear case walls recorded for an unworn gear. The essence of the method is the determination of the frequency of vibrations characteristic for individual damage types. In case of toothed gears the most characteristic features include meshing frequency $f_{\mathrm{z}}$ (Eq. 2):

$$f_{\mathrm{z}} = \frac{n_1 \cdot z_1}{60} \tag{2}$$

and rotational frequencies of shafts $f_{\mathrm{w1}}$ (3), $f_{\mathrm{w2}}$ (4):

$$f_{\mathrm{w1}} = \frac{n_1}{60} \tag{3}$$

$$f_{\mathrm{w2}} = \frac{n_2}{60} = \frac{n_1}{60} \cdot \frac{z_1}{z_2} \ . \tag{4}$$

Damage of rolling bearings manifests itself in vibrations the frequencies of which depend on the bearing design and the rotational speed of the shaft. For each bearing there are four characteristic frequencies corresponding to the damage of: outer ring, inner ring, rolling part and bearing cage, respectively defined in the subject industrial catalogues. One of the basic parameters which has an impact on the operation of a gear is temperature therefore it is necessary to record the temperature changes as a function of time. The temperature of mating parts increases as the load and peripheral speed increase. The basic heat sources in a gear include temperature on the side face of the mating wheel teeth and the temperature on the bearing housing. Large quantities of heat is produced at high speed and pressure which results in lower viscosity of the lubricant and, thus, thinner oil film or even lack of it. Temperatures are observed on the actual elementary contact surfaces which lead to softening or even partial melting of metal. Thermocouples of J-type were used for temperature measurements. They were connected to the controller via specialized thermocouple module THM 889. Recording of rotational speed on the shaft was done with the use of incremental encoder fixed to the motor housing and the fast counter module APU 300. Tests carried out for some time at the above mentioned stand (Fig. 3) confirm that the set of signals collected by the control system can prove sufficient to detect possible damage. The use of such a system may contribute to early damage detection and, consequently, to prevent further deterioration of the machine. Next stage of development of the control system presented is to use it for diagnosis of a number of devices where the measuring signals are collected via distributed control system and transmitted to a PC equipped with implemented diagnostic algorithms for each of the parts to be diagnosed (Fig. 6).

Acquisition of inputs signals and control over a single stand are performed by a PLC which is a node in a distributed system. Testing of a distributed system consists in determination of delays occurred during communication between the client (PC) and the server (PLC). Determination of delays in a distributed control system enables to determine the system response time to a hazard present during operation and it is also required to determine the size of memory buffers and of the data set being sent, necessary to carry out numerical analysis.

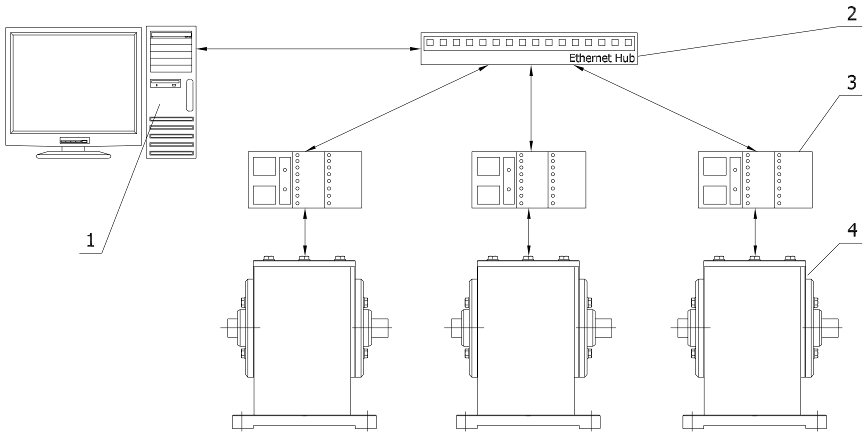



**Fig. 6.** Developed control and diagnostic system: 1 – central unit, 2 – Ethernet hub, 3 – Ethernet interface (1 . . . 16), 4 – gear box

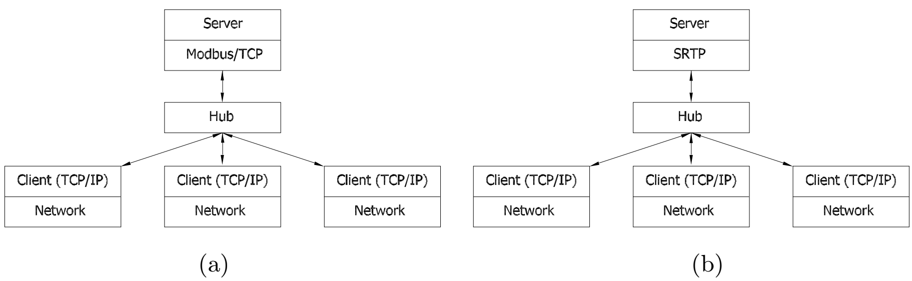



**Fig. 7.** Configuration for: (a) MODBUS/TCP, (b) SRTP

The subject distributed system uses Ethernet as a basic communication interface and a stack of TCP/IP protocols. It is known that the collision detection and response mechanism does not guarantee the time determinism. Taking the control over data exchange in Ethernet by another deterministic transmission protocol enables to configure the network in such a way that the determinism condition is met. Two configurations of the communication system for the stand in question were assumed for the analysis of the solution. First configuration where performed for MODBUS/TCP protocol and second for SRTP protocol (Fig. 7).

## 3 Data Exchange Model for MODBUS/TCP Protocol and SRTP Protocol

The major objective, under the tests carried out, was to determine a communication system model and its impact on the control system. This chapter presents a basic data transaction exchange model for triggered and cyclic exchanges for SRTP protocol based on Ethernet [8, 9].

The relation (5) describes duration of a single exchange of data reading or writing

$$T_{\mathrm{WW}} = T_{\mathrm{ZM}} + T_{\mathrm{PM}} + 18\ (T_{\mathrm{PR}} + T_{\mathrm{TR}} + T_{\mathrm{DR}} + T_{\mathrm{AR}}) + T_{\mathrm{AS}} \tag{5}$$

where:

$T_{\mathrm{WW}}$ – time for performance of exchange triggered; query/instruction-response for Ethernet with SRTP protocol;
$T_{\mathrm{ZM}}$ – time measured from the moment of reporting the exchange during cycle stage ($T_{AP}$) to the start of communication stage ($T_K$) in CLIENT station;
$T_{\mathrm{AS}}$ – duration of SERVER station cycle;
$T_{\mathrm{PM}}$ – time from decoding by the CLIENT station co-processor of the piece of information to its being accepted and used in an application;
$T_{\mathrm{PR}}$ – frame preparation time;
$T_{\mathrm{TR}}$ – frame transmission time;
$T_{\mathrm{DR}}$ – frame detection time;
$T_{\mathrm{AR}}$ – frame analysis time;
18 – constant value measured on the basis of own experimental tests which define the number of messages sent within the network in the form of separate frames.

The MODBUS/TCP protocol consist four different types of input/output operations: write, read, open port and close port. Each open of port command is called from application level and consist information about type of device with which transaction be able. All of communication messages are called from application level of Programmable Logic Controllers. If connection with other device must be done, open of communication port and using of read/write function, is necessary. After communication process, port must be closed. Time of open port for MODBUS/TCP protocol, describes relation (6).

$$T_{\mathrm{MOP}} = T_{\mathrm{ZM}} + T_{\mathrm{PM}} + 3\ (T_{\mathrm{PR}} + T_{\mathrm{TR}} + T_{\mathrm{DR}} + T_{\mathrm{AR}}) \tag{6}$$

where:

$T_{\mathrm{MOP}}$ – time of port opening;
3 – constant value measured on the basis of own experimental tests which define the number of messages sent within the network in the form of separate frames.

For closing port:

$$T_{\mathrm{MZP}} = T_{\mathrm{ZM}} + T_{\mathrm{PM}} + 4\ (T_{\mathrm{PR}} + T_{\mathrm{TR}} + T_{\mathrm{DR}} + T_{\mathrm{AR}}) + T_{\mathrm{AS}} \tag{7}$$

where:

$T_{\mathrm{MZP}}$ – time of port closing;

4 – constant value measured on the basis of own experimental tests which define the number of messages sent within the network in the form of separate frames.

For exchange data operations, write/read functions:

$$T_{\mathrm{MZO}} = T_{\mathrm{ZM}} + T_{\mathrm{PM}} + 4\ (T_{\mathrm{PR}} + T_{\mathrm{TR}} + T_{\mathrm{DR}} + T_{\mathrm{AR}}) + T_{\mathrm{AS}} \tag{8}$$

where:

$T_{\mathrm{MZO}}$ – time of read or write transaction.

Total value for all operations – time of full exchange transaction for MODBUS/TCP protocol, port opening, write or read, closing port:

$$T_{\mathrm{WM}} = T_{\mathrm{MOP}} + T_{\mathrm{MZO}} + T_{\mathrm{MZP}}\ . \tag{9}$$

## 4 Experimental Test

At the beginning of experimental test, decided that control and regulation system will be monitored. Monitored system was made on Ethernet networks. During tests, the measurements of time transaction were made. Transaction time measurement was performed on a stand designed in Client-Server system. Data exchange transaction duration was measured for the following exchange scenarios:

- scenario 1
  - reading data from client station 1 for MODBUS/TCP,
  - writing data to client station 1 for MODBUS/TCP,
- scenario 2
  - reading data from client station 1 for SRTP,
  - writing data to client station 1 for SRTP.

## 5 Summary

During experimental tests made a measurements of transactions read/write time, for two protocols: SRTP and MODBUS/TCP. The measurements were made for shortest PLC cycle. Thanks that exchange data time was measured for different data blocks. For each data exchange made 1,000 probes. From all of results, the maximum and minimum value of time was find. The results are on Fig. 8.

The proposed solutions of control and regulation system can be used for control and monitoring damages symptoms of mechanical systems. For very fast diagnosis of damaged element, the registration of typical symptoms is necessary, for compare with good one. Is also necessary make some diagnosis tests for method verifications and make of data base for diagnostic process of mechanical devices.

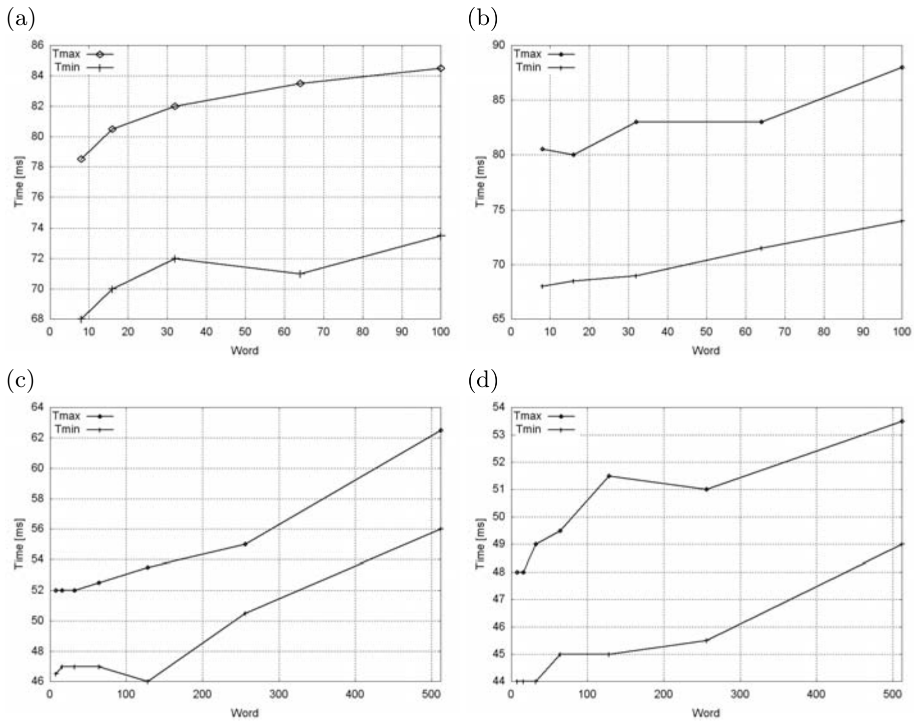



**Fig. 8.** Reasults of measurements for MODBUS/TCP: (a) read, (b) write; and SRTP: (c) read, (d) write

## References


1. Kwiecień, A.: Analiza przepływu informacji w komputerowych sieciach przemysłowych (in Polish). Studia Informatica, Gliwice (2002)
2. Kwiecień, A., Sidzina, M.: The programmable logic controller PLC work cycle time influence on exchange data time in industrial networks (in Polish). In: AUTOMATION 2004, Warszawa, pp. 593–601 (2004)
3. Kwiecień, A., Rysinski, J., Sidzina, M.: Analiza wymiany danych w sieci Ethernet dla sterowników swobodnie programowalnych. In: Kwiecień, A., Gaj, P. (eds.) Systemy informatyczne z ograniczeniami czasowymi (in Polish), pp. 233–246. WKŁ, Warszawa (2006)
4. Klosinski, J., Rysinski, J., Sidzina, M.: Experimental investigation of load alterations of gears using fuzzy-logic controller. Acta Mechanica Slovakia R11 4-A, 179–186 (2007)
5. Rysinski, J., Sidzina, M.: Measurements of gear temperature distribution by means of free-programmable controllers. Acta Mechanica Slovakia R10 4-B, 295–304 (2006)
6. Mirota, K., Rysinski, J., Tomaszewski, J.: Diagnostics of heat generation process in cylindrical gears. In: VIII International Scientific Conference, Trencin, pp. 408–415 (2006)

7. Mirota, K., Rysinski, J., Tomaszewski, J.: Diagnostics of scoring in gears. Solid State Phenomena 144, 118–123 (2008)
8. Sidzina, M.: Badanie przepływu informacji z sieci MASTER-SLAVE do sieci TOKEN-RING z potwierdzeniem lub bez. Zeszyty Naukowe. Budowa i Eksploatacja Maszyn, Akademia Techniczno-Humanistyczna, Bielsko-Biała, 128–135 (2004) (in Polish)
9. Sidzina, M.: Dynamiczne modyfikacje programu aplikacji sterownika swobodnie programowalnego celem zwiększenia częstości wymian komunikatów w przemysłowych systemach rozproszonych czasu rzeczywistego. Rozprawa doktorska, Politechnika Śląska Wydział Automatyki Elektroniki i Informatyki, Gliwice (2008)

# Improving Availability of Industrial Monitoring Systems through Direct Database Access

Arkadiusz Jestratjew

Silesian University of Technology, Institute of Informatics,
Akademicka 16, 44-101 Gliwice, Poland
arkadiusz.jestratjew@polsl.pl

**Abstract.** From an architectural point of view, a typical industrial process monitoring system consists of three main components: a data source (i.e. process controller), a data store and an intermediary communication infrastructure. The paper covers monitoring systems where communication infrastructure is based on a gateway PC. The gateway is running dedicated communication software to retrieve process data and store it into the database. An alternative approach is proposed to improve availability of the monitoring system through using direct database access and thus avoiding the gateway PC usage.

## 1 Introduction

In a global economy and strong competition world, enterprises are encouraged to provide its customers with low cost and high quality goods. In order to achieve that goal, production processes are continuously monitored for any deviations of process parameters. Monitoring systems allow detecting problems early before they arise and make possible applying proper preventive or correction actions.

Furthermore, parameters of some safety-related process operations (e.g. assembling of suspension elements into a car body) or even a whole production processes (e.g. steel melting) have to be recorded for each produced item. Thus, when monitoring system is not available, production process either have to be stopped or produced goods are considered scrap if the process cannot be stopped. That makes availability a critical factor for these monitoring systems.

Typical process monitoring system (Fig. 1) consists of the following three main components: a *data source* (i.e. process controller), an *intermediary communication infrastructure* and a *data store*.

In the paper, an *availability* of an industrial monitoring system is defined as its *readiness* to recording data provided by the data source without any data loss. Thus, data source failure does not affect availability of the monitoring system. However it does affect the entire process availability, as process controller failure does likely stop the process.

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2009, CCIS 39, pp. 344–351, 2009.

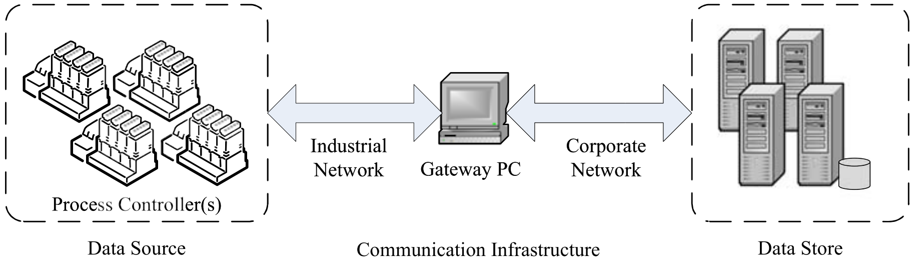



**Fig. 1.** Architecture of an industrial process monitoring system

# 2 Industrial Monitoring System Components

## 2.1 Data Source

Automatic production processes are controlled by various computational devices, like *Programmable Logic Controllers* (PLC, e.g. [22]), *Distributed Control Systems* (DCS, e.g. [1]) or powerful *Programmable Automation Controllers* (PAC, e.g. [5]). These devices are equipped with specialized input/output hardware that enable both process data acquisition from sensors and process control through actuators. *Process controller* is used as a general term to name that kind of devices through this paper.

Process controllers are devices specifically designed to handle interferences encountered in industrial plants, like dust, vibrations, electromagnetic interference etc. and still function reliable for years. Even higher availability and reliability is achieved with hardware redundancy that is supported by high-end process controllers [6,26].

Process controllers are natural data sources for monitoring systems, as all process data is processed by them. Considering industrial monitoring systems it is possible that separate devices perform process data acquisition and actual process control. In such case, the device that performs data acquisition function is considered a data source.

## 2.2 Data Store

Data store is a part of the back-end corporate IT infrastructure. Improving availability and scalability of such infrastructure in business-critical applications has a long history. Nowadays, multiple products and technologies are commercially available, including database server clustering, connection brokers, load balancers, application server farms and hardware redundancy, to name just a few of them. These technologies together with physical security, controlled environmental conditions and valid operation procedures allow building back-end systems that fulfils corporate requirements of availability, scalability and performance [7,13]. Figure 2 depicts example architecture of such system.

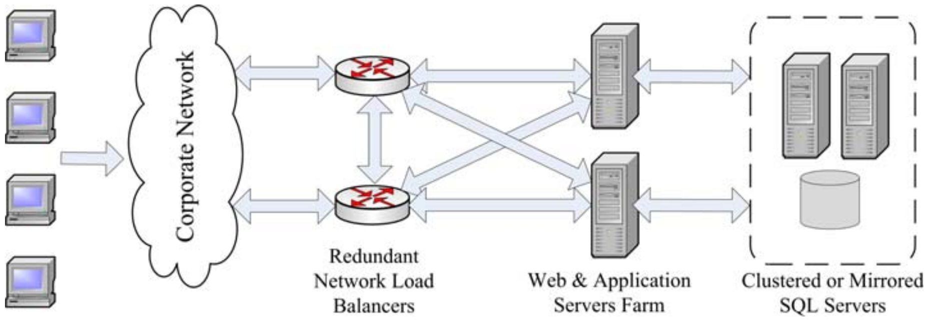



**Fig. 2.** Back-end corporate data store designed for high availability, based on [13]

In context of industrial monitoring systems, it shall be considered that raising data store availability to required level is more a cost-to-benefit problem than engineering problem.

### 2.3 Communication Infrastructure

Communication infrastructure in typical monitoring system consists of the following three subcomponents.

1. Industrial network with dedicated communication protocol.
2. Corporate network that provides access to corporate back-end data store infrastructure mentioned earlier, typically over a fast Ethernet connection.
3. Gateway device that bridges both worlds, execute protocol stacks and perform necessary data conversion.

Industrial networks are designed to work reliably in a hard environment where multiple sources of electromagnetic interference exist. Accordingly many industrial networks use physical layer that is incompatible with standard communication interfaces available in PC computers (e.g. Profibus [9], CAN [10] or RS-485 [29]) and thus specialized hardware converters must be used. Converters are built either as stand-alone devices connected with a standard RS-232 or USB serial port, or as a specialized PC computer extension card. Such an extension card may have an on-board processor that fulfils industrial network protocol on its own, especially for fast networks like CAN [2] or Profibus [27].

### 2.4 The Gateway

Depending of data exchange configuration and underlying communication protocol, a process running on the gateway device (Fig. 3) either actively requests, or gets notified, over the industrial network, about data changes in the monitored process. Next, gathered data is persisted into the data store using corporate network and standard protocols, either HTTP-based (SOAP [28], REST [3,4]) or binary (Sybase/Microsoft TDS [14], Oracle SQL*Net, etc.).

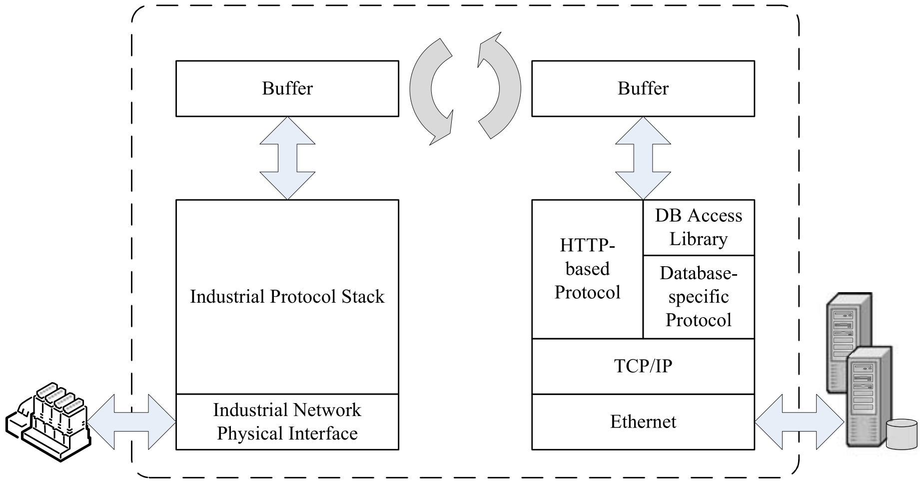



**Fig. 3.** Logical view of the gateway

Typical gateway device is an industrial-class PC computer, located nearby the process controller. It is exposed on hard environmental conditions in production plants, like dust, vibrations, electromagnetic interferences and voltage spikes on power lines. Comparing to safe environment in air-conditioned server rooms, one may suppose that even hardened industrial-class gateway PC is subject to hardware failure with relatively high probability. Gateway PCs are also sometimes used as process visualization devices. That implies continuous user interaction and may lead to increased probability of failure because of improper operator activities.

Unfortunately, the gateway PC is a single point failure that stops entire monitoring system in case of hardware or software error. Although it might be possible to use redundant gateway devices, such solution ultimately complicates protocol converter software. When the gateway PC acts as a *Master* node in Master-Slave industrial networks, which is a common case, it may be even impossible to detect a failure and perform switch-over to redundant gateway while time constraints are still held. On the other hand, redundancy in industrial network protocols does not come for free, in term of both cost and time [18,19] and thus shall be avoided when possible.

## 3 Direct Database Access in Monitoring Systems

Many modern process controllers are equipped with Ethernet interface, simplifying development of industrial monitoring systems, as physical layer converters are no longer necessary. In such environment, a gateway PC is used mainly as application layer protocol converter. Avoiding usage of the gateway PC will increase availability of the monitoring system with no additional cost related with redundancy, as single point of failure is no longer present.

Industrial network protocols (e.g. Modbus-TCP [15], Ethernet Global Data [11], Saia S-Bus [25], etc.) are incompatible with data store access protocols mentioned above (e.g. TDS or HTTP), despite these protocols are all based on TCP/IP protocol stack.

To achieve interoperability, data store access must be handled inside data source. The opposite case, i.e. integrating the gateway functionality into the data store, is likely out of concern, because of operating procedures, corporate politics or technical problems that will arise, including, but not limited to, the following:

- hardware platform or operating system incompatibilities,
- additional software development and test complexity and cost associated with server runtime environment,
- additional workload generated by, possibly multiple, protocol converters,
- tight coupling of Ethernet-based industrial network and corporate network,
- future problems during data store migration to a new platform, as process control systems lifetime is significantly longer that typical IT infrastructure lifetime, etc.

Implementing data store access protocols within process controller, the data store may be considered as a black box accessible by either dedicated binary protocols or HTTP-based protocols. However, it is required that firmware of the process controller supports implementing custom protocols by application software. Such functionality is common in case of serial port communications, by means of so-called 'character mode' [5,22] allowing application software to directly process data sent and received over a serial port. Unfortunately, in case of Ethernet port communications, similar functionality is rather uncommon. Saia-Burgess PCD series PLCs are an example of devices with firmware support of direct access to TCP or UDP sockets by application software [23]. However, it shall be expected that future developed devices or firmware updates will bring that functionality to the market.

Using binary protocols may provide higher performance, however as these protocols are proprietary and may be protected with patents, application developer shall be aware of additional costs of licence fees. Moreover, using binary protocol restricts supported database system to specific vendor and limits future enhancements of corporate infrastructure.

Using HTTP-based rather than binary protocols to access the data store provide much more flexibility, and thus shall be preferred. However, implementing HTTP client in process controller application software is not a trivial task, mainly because commonly used programming languages are designed to process logic equations and numeric data rather that strings that are extensively used in HTTP [20].

## 4 Performance Considerations

In order to estimate execution costs of direct database access over HTTP-based protocol, a simple HTTP client code was developed for Saia-Burgess

PCD3.M5540 programmable logic controller using Instruction List programming language. To avoid confusion it shall be noted that PCD Series Instruction List [24] and Instruction List defined by IEC 61131-3 [8], while sharing some concepts, are completely unrelated programming languages, despite the name.

Data store was built on Intel Pentium 4-based PC with 1024 MB of RAM running Windows 2003 Server Standard Edition Service Pack 2 and Microsoft SQL Server 2005 Express Edition database server. Microsoft ASP.NET Data Services Framework (formerly code name 'Astoria') [12] was used to provide access to the database over HTTP protocol with a simple, REST-based programming model and JSON [21] data encoding. REST and JSON were chosen in order to possibly reduce required string processing operations.

During the experiment, requests were issued at approximate rate of two to three requests per second. A request consists of multiple data items contained within HTTP POST request. HTTP response contains 'HTTP 201 Created' success code together with created entity data, and is ignored. Multiple measurements of processing time vs. data item count were taken. JSON serialization time, HTTP request preparation time and low-level TCP/IP processing time were measured independently. Results of measurements are shown on Fig. 4.

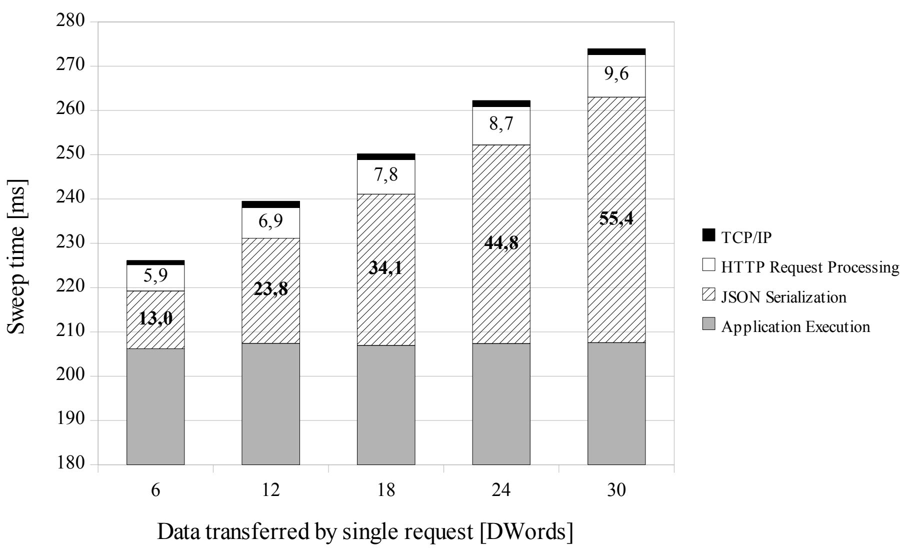



**Fig. 4.** Average HTTP-client software processing time

## 5 Alternatives

OPC Unified Architecture (UA) specification developed by the OPC Foundation might be seen as an alternative solution to the problem mentioned above. As described in [16] 'OPC UA is a platform-independent standard through which various kinds of systems and devices can communicate by sending *Messages* between *Clients* and *Servers* over various types of networks'. OPC UA currently supports TCP and SOAP over HTTP based communication with either binary or XML data encoding, to provide flexibility for data transfer 'from plant floor PLCs to enterprise servers' [16].

In the OPC UA environment, achieving high availability is possible by client and server redundancy. OPC UA specification defines data structures and services by which redundancy may be achieved in a standard manner, both for client redundancy and transparent or non-transparent server redundancy [17]. Such redundancy support may overcome availability problems for the Gateway PC if both Data Source and Gateway PC support OPC UA.

Unfortunately, wide adoption of OPC UA is a long-term process. Meantime, proposed direct database access architecture and OPC UA may be seen as complementary technologies, especially for currently available process controllers that do not support OPC UA directly.

## 6 Conclusions

Elimination of the gateway PC from industrial monitoring systems as shown on Fig. 1 seems to be a way to increase monitoring system availability. As typical industrial network protocols are incompatible with database access protocols, it is required to implement one within application software of the controller. To achieve highly flexible and loosely coupled monitoring system architecture, it is proposed to use HTTP-based communication protocols, as REST or SOAP.

Unfortunately, creating custom communication protocols within application software for Ethernet-based networks is rarely supported by controller vendors. Moreover, implementing custom HTTP client encounter difficulties in both code complexity and execution time because programming languages that are available are rather inadequate for that task. As a result, application area of the proposed approach is not as wide as it could be. However, HTTP-based communication is proven to be possible and might be useful is some scenarios. Above problems shall be resolved by controller vendors, by implementing in the firmware a fully featured HTTP client besides HTTP server that is often currently implemented.

## References

1. ABB Automation Technologies: Industrial$^{\mathrm{IT}}$ System 800xA System Architecture Overview (2005)
2. Beckhoff GmbH: CANopen PCI Card FC5101 and FC5102 Manual (2002)
3. Fielding, R.T.: Architectural Styles and the Design of Network-based Software Architectures. PhD thesis, University of California, Irvine (2000)
4. Fielding, R.T., Taylor, R.N.: Principled design of the modern Web architecture. ACM Trans. on Internet Technology 2(2), 115–150 (2002)
5. GE Fanuc Automation: PACSystems$^{\mathrm{TM}}$CPU Reference Manual. Document GFK-2222K (2007)

6. GE Fanuc Automation: PACSystems$^{TM}$Hot Standby CPU Redundancy. Document GFK-2308A (2006)
7. Hirt, A.: Pro SQL Server 2005 High Availability. Apress (2007)
8. IEC 61131:2003 Programmable controllers – Part 3: Programming languages. International Standard (2003)
9. IEC 61158 Industrial communication networks – Fieldbus specifications, Type 3. International Standard (2007)
10. ISO 11898-1:2003 Road vehicles – Controller area network (CAN) – Part 1: Data link layer and physical signalling. International Standard (2003)
11. MacLeod, D.: Computer Communications for Ethernet Global Data – CommEGD. GE Fanuc Automation Support Site (2000)
12. Microsoft Corp.: ADO.NET Data Services Framework. MSDN Library
13. Microsoft Corp.: Microsoft High Availability Overview White Paper (2008)
14. Microsoft Corp.: Tabular Data Stream Protocol Specification. MSDN Library
15. Modbus IDA: MODBUS Messaging on TCP/IP Implementation Guide (2006)
16. OPC Foundation: OPC Unified Architecture Specification. Part 1: Concepts (2006)
17. OPC Foundation: OPC Unified Architecture Specification. Part 4: Services (2007)
18. fKwiecień, A., Stój, J.: The Response Time of a Control System with Communication Link Redundancy. In: Węgrzyn, S., Czachórski, T., Kwiecień, A. (eds.) Contemporary Aspects of Computer Networks, vol. 2, pp. 195–202. WKŁ, Warszawa (2008)
19. Stój, J.: Wpływ redundancji na zależności czasowe w rozproszonych informatycznych systemach czasu rzeczywistego. PhD thesis, Silesian University of Technology, Gliwice (2009) (in Polish)
20. RFC2616 Hypertext Transfer Protocol – HTTP/1.1. Internet Engineering Task Force, The Internet Society (1999)
21. RFC4627 The application/json Media Type for JavaScript Object Notation (JSON). Internet Engineering Task Force, The Internet Society (2006)
22. Saia-Burgess Controls Ltd.: Hardware Manual for the PCD3 Series. Document 26/789 Edition E8, Murten (2007)
23. Saia-Burgess Controls Ltd.: SAIA PCD Controllers with Ethernet-TCP/IP Manual. Document 26/776 Edition E3, Murten (2003)
24. Saia-Burgess Controls Ltd.: SAIA PCD Reference Guide. Document 26/733 Edition E6, Murten (2001)
25. Saia-Burgess Controls Ltd.: S-Bus Manual for the PCD family. Document 26/739 Edition E4, Murten (2000)
26. Siemens, A.G.: Automation System S7-400H – Fault-tolerant Systems. Document 6ES7988-8HA11-8BA0 (2006)
27. Siemens, A.G.: SIMATIC NET CP 5621 Operating Instructions. Document A5E0-1156749 (2008)
28. SOAP Version 1.2, 2nd edn. W3C Recommendation (2007)
29. TIA-485-A Electrical Characteristics of Generators and Receivers for Use in Balanced Digital Multipoint Systems. ANSI/TIA/EIA-485-A-98, R2003 (1998)

# Author Index