

Mechanisms and Machine Science 42

Stanisław Zawiślak
Jacek Rysiński *Editors*

Graph-Based Modelling in Engineering

 Springer

Mechanisms and Machine Science

Volume 42

Series editor

Marco Ceccarelli

LARM: Laboratory of Robotics and Mechatronics

DICeM: University of Cassino and South Latium

Via Di Biasio 43, 03043 Cassino (Fr), Italy

e-mail: ceccarelli@unicas.it

More information about this series at <http://www.springer.com/series/8779>

Stanisław Zawiślak · Jacek Rysiński
Editors

Graph-Based Modelling in Engineering

 Springer

Editors

Stanisław Zawiślak
Faculty of Mechanical Engineering and
Computer Science
University of Bielsko-Biala
Bielsko-Biała
Poland

Jacek Rysiński
University of Bielsko-Biala
Bielsko-Biała
Poland

ISSN 2211-0984

Mechanisms and Machine Science

ISBN 978-3-319-39018-5

DOI 10.1007/978-3-319-39020-8

ISSN 2211-0992 (electronic)

ISBN 978-3-319-39020-8 (eBook)

Library of Congress Control Number: 2016942487

© Springer International Publishing Switzerland 2017

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG Switzerland

Preface

This book is dedicated to some chosen aspects of an application of graph theory in some fields of engineering. There are plenty of books concerning graph theory (treated from a theoretical point of view) on the worldwide market; however, its application is not too frequently published in a form of separate volumes. In fact, among all recently published books related to the field of graph theory, discrete mathematics and combinatorics, there is probably only a few per cent dedicated to graph application.

On the other hand, there are conference proceedings dedicated to applied or industrial mathematics where graphs' applications are mixed with versatile other mathematical tools suitable for modelling—as for example: differential and integral equations, algebraic equations or geometrical 3D models. The applications of graph theory are versatile, e.g. in chemistry, electrical engineering, network analysis, mechanical engineering, automation, computer science and especially nowadays social sciences as well as social networks modelling and analysis. The very recent book printed by SPRINGER (in 2015), entitled “Graph-Based Representations in Pattern Recognition” is dedicated to the idea of merging graphs with artificial intelligence-related problems.

The present book gives a glimpse on some graph theory application in modelling networks in general and modeling chosen mechanical systems as for example: robot leg, planetary gears and compound gear system consisting of planetary and friction gear. Moreover, some applications of graphs in modelling of production processes as well as design processes are included. Chemistry and electrical engineering topics are not considered at all, but it would be possible in next editions.

A model of a particular system is obtained via a series of activities, e.g. abstraction, simplification, discretization and derivation of relations among the elements of the system. Therefore, the graph models do not represent the system in a holistic way, but on contrary, some of its features or behaviours are grasped and other omitted. Therefore, some approaches are neglected, obviously; however, it allows focusing an attention on some other important aspects.

This book contains also some basics dedicated to bond graphs which are recently most popular due to their generality and possibility to capture different subsystems (mechanical, electrical, hydraulic etc.) in one unified model. Moreover, professional and shareware-type software is available to create the graph-based models for simulation of behaviour of the systems. Therefore, the bond graphs are widely developed and taught at the universities, especially in Holland, Romania and USA as well as Poland.

Besides bond graphs, other types of graphs are utilized in the present book of gathered papers, e.g. Petri nets, contour graphs, linear, directed and mixed graphs as well as logical trees. Despite the fact that the book contains just 18 contributions, it gives a wide spectrum of graph application. Additionally, due to the wide lists of references and based on the contents itself, it allows for understanding the state of art in contemporary graph-based modelling of wide range of systems, networks and engineering tasks as design and decision-making. Graphs are powerful and handy tools for engineers, and number of papers dedicated to their utilization has been increasing in recent years essentially; however, monographs are really rare. We expect that this book would launch the sub-series dedicated to graph theory applications.

This book is included into the series “Mechanisms and Machine Science” edited by Prof. Marco Ceccarelli (Casino, Italy) under patronage of IFToMM (The International Federation for the Promotion of Mechanism and Machine Science). The mission of IFToMM is developing not only the mechanism and machine science itself but also any utilization of modelling tools; moreover, IFToMM propagates multidisciplinary approach, as well.

We do hope that this book would be an inspiration for studying graph theory taking into account its versatile application.

Bielsko-Biała, Poland
April 2016

Stanisław Zawiślak
Jacek Rysiński

Contents

Part I Basics and Theoretical Aspects of Graph and Network Modelling	
Bond Graphs in System Modelling	3
R. Ibănescu	
Structural Importance and Local Importance in Network Reliability	25
P. Tittmann and S. Kischnick	
Access Distribution Scheme to the Computer System Based on Fuzzy Logic	39
A. Shaikhanova, A. Zolotov, L. Dubchak, M. Karpinski and V. Karpinskyi	
Problem of Medicines Distribution on the Example of Pharmaceutical Wholesale	51
M. Cieśla and B. Mrówczyńska	
Part II Modelling of Mechanical Systems	
Bond Graph Model of a Robot Leg	69
B. Gola, J. Kopec, J. Rysiński and S. Zawislak	
Kinematical Analysis of Variants of Wind Turbine Drive by Means of Graphs	81
J. Drewniak, J. Kopeć and S. Zawiślak	
Graph-Based Algorithm for the Evaluation of the Mechanical Efficiency of Epicyclic Gear Drive in Hybrid Scooters	97
P. Coaccioli and E. Pennestri	
Graph-Theoretic Modelling and Sensitivity Analysis of Dynamic Systems	107
J. Banerjee and J. McPhee	

Three-Dimensional Analysis of Vehicle Stability Using Graph Theory	117
G.G. Moreno, R.L.P. Barreto, R.S. Vieira, L. Nicolazzi and D. Martins	
Analysis of the Kinematics of Planar Link Mechanism with Non-stationary Motion of Crank	131
J. Drewniak, P. Garlicka and B. Borowik	
Part III Modeling of Production Processes	
Petri Nets for Computer Aided Group Technology	143
R. Strzycek	
Graph Theory in Product Development Planning	165
I. Kutschenreiter-Praszkiewicz	
A Digital Pattern Approach to 3D CAD Modelling of Automotive Car Door Assembly by Using Directed Graphs	175
S. Patalano, F. Vitolo and A. Lanzotti	
Part IV Graph-Based Aid and Modelling of Design Tasks	
Application of Game Graphs to Describe the Inverse Problem in the Designing of Mechatronic Vibrating Systems	189
A. Deptuła	
Graphic Matrix Formalization of Logical Decision Trees in the Optimization of Machine Systems	201
A. Deptuła and M.A. Partyka	
The Class of Objects Graph Model as Dataware of Structural Synthesis System	211
O.V. Malina and E.G. Zarifullina	
Part V Miscellaneous	
Search Module as a Tool for Improvement of Classifier	223
E.G. Zarifullina, O.V. Malina and I.M. Nekipelova	
Kazimierz Kuratowski—Biography and Genesis of the Theorem on Planar Graphs	233
J. Wojnarowski and S. Zawiślak	
Author Index	247

Part I
**Basics and Theoretical Aspects of Graph
and Network Modelling**

Bond Graphs in System Modelling

R. Ibănescu

Abstract The paper focuses on the presentation of a graphical modeling method named a bond graph method that is applicable to various dynamic systems such as mechanical, electrical, hydraulic, thermal, chemical and magnetic or a combination of these named hybrid systems or multidisciplinary systems. The method is based on an analysis of the power flow in a system from the power sources to its working elements. A brief presentation of the method illustrates its basic concepts and intends to outline the ease and the benefits of using it and its huge potential for multidisciplinary systems modelling, as well.

Keywords Bond graph · Mathematical model · Block diagram model · Hybrid systems

1 Introduction

A frequently used procedure in systems design and analysis consists in constructing a mathematical model. There are different methods to obtain such a model. Unfortunately, there is no rule to choose a method or another technique, for this reason, it is the engineer's decision, according to her/his experience and abilities. All these methods have in common that they lead directly and immediately to a system of ordinary differential equations (ODE) or, worse, to a system of algebraic-differential system of equations (DAE). Some methods, such as Lagrange's equations method, are a sort of 'black box' method, which gives little insight into the physical phenomena that lies behind the functioning of a particular system. It becomes more and more clear that the power exchanges between the parts of a device play the most important role in system analysis and modeling. This aspect urged scientists to find a new approach to system modeling, in order to exploit power transfer. In the middle of the last century, Paynter completed a new

R. Ibănescu (✉)

Technical University "Gheorghe Asachi" of Iași, Iași, Romania
e-mail: ribanesc@tcm.tuiasi.ro

modeling method, based on a power exchange, called a Bond Graph [1]. The birthday of this method is considered to be April 24, 1959, as the author himself declared. This new modeling method has developed very quickly, due to its great advantages and it became a target of scientific work for many researchers [2–18]. What can be more attractive than a graphical method, suitable for various energetic domains (mechanical, electrical, hydraulic, thermal etc. or a combination of them, forming hybrid systems [18–23]), based on a thorough analysis of a power exchange between system elements and providing directly a simulation scheme, suitable for appropriate software, without writing any equations? These advantages determined a quite important part of the scientific world to focus on the bond graph method and to improve its capabilities. Nowadays, the bond graph methodology is extended to many fields, such as nuclear reactors [24, 25], optimal control [26], bioengineering [27–29], mechanisms and machines [30–33], environment engineering [34], chemical industry [35], switching systems [36], uncertain models [37], robotics [38], population interactions [39], renewable energy [40–45], genetic programming [46], estimation and control [47], acoustics [48], magnetic systems [49], identifiability [50], finite element [19], fault diagnosis [51], statics of systems [52, 53], theory of sonics [54] etc. On the other hand, a lot of dedicated software has been created, in order to directly implement the bond graph model in computers. The present work is organized as follows: Sect. 2—About Power and Energy Variables; Sect. 3—The Bond Graph Elements; Sect. 4—Causality in Bond Graphs; Sect. 5—Construction of the Bond Graph; Sect. 6—Exploitation of the Bond Graph as a Meta Model; Sect. 7—An Illustrative Exemplary Use of the Bond Graph Method for a Hydro-Mechanical System; Sect. 8—Conclusions.

2 About Power and Energy Variables

It is well known that every moving system has a power source. The idea is to consider the power as a single physical quantity that causes the system to function and then, to analyze the power as the result of the simultaneous presence of the two physical quantities, whose product is power. Let's consider, for example, a socket as a power source. The power is given as the product of voltage and current. The voltage is constant, but the current varies, depending on the necessary power. For example, a bulb will need a smaller current than a heating device. This point of view lets one consider the voltage as a *cause* physical quantity and the current as an *effect* physical quantity. More than that, the socket can be considered as an *ideal source of power*, as long as the maximum power it can provide is not exceeded. The socket is more specifically named an ideal source of voltage, because the voltage is the cause. Of course, there are also ideal sources of current. There are ideal sources of power in any other usual energetic field and one of the physical quantities of power will always be the cause, while the other one will represent the effect.

In the bond graph methodology, the two components of power (denoted $P(t)$) are generally named *effort* (denoted e) and *flow* (denoted f), one being always the cause

Table 1 The classical correspondence of effort and flow in the main energetic fields

Energetic field	Effort e	Flow f
Mechanical translation field	Force F [N]	Velocity v [m/s]
Mechanical rotation field	Torque M [Nm]	Angular velocity ω [s^{-1}]
Electrical field	Voltage u [V]	Current i [A]
Hydraulic field	Pressure P [Pa]	Volume flow rate Q [m^3/s]
Thermal field	Temperature T [K]	Entropy flow rate \dot{S} [J/Ks]

Table 2 The classical correspondence of generalized momentum and generalized displacement in the main energetic fields

Energetic field	Generalized momentum p	Generalized displacement q
Mechanical translation field	Momentum p [Ns]	Linear displacement x [m]
Mechanical rotation field	Angular momentum L [Nms]	Angular displacement θ [rad]
Electrical field	Magnetic flux Φ [Wb]	Electrical charge q [C]
Hydraulic field	Pressure momentum p_P [Ns/m ²]	Volume V [m ³]
Thermal field	–	Entropy S [J/K]

and the other being the effect ($P(t) = f(t)e(t)$). They are called *power variables*. The correspondence of the effort and the flow in mechanical, electrical, hydraulic and thermal fields, from the classical point of view, are shown in Table 1.

Another important physical quantity is the energy stored in some elements of a system. The stored energy can be expressed by using two kinds of variables, named *energy variables*. They are also the state variables in the bond graph methodology. These variables are the *generalized momentum* (denoted by p) and the *generalized displacement* (denoted by q).

The correspondence of the generalized momentum and the generalized displacement in mechanical, electrical, hydraulic and thermal fields, from the classical point of view, is shown in Table 2.

The following relations exist between the power variables and energy variables:

$$\dot{q}(t) = f(t) \quad \dot{p}(t) = e(t)$$

$$q(t) = \int_0^t f(\tau) d\tau + q(t_0) \quad p(t) = \int_0^t e(\tau) d\tau + p(t_0). \quad (1)$$

For an easier and more intuitive use, the *mass flow* (\dot{m} [kg/s]), the *heat flow* (\dot{Q} [J/s]) or the *enthalpy flow* (\dot{H} [J/s]) may be used, instead of *volume flow* or *entropy flow*. In this case, the bond graph model will be called a *pseudo bond graph*, because the product $e(t) \cdot f(t)$ is no longer a power. The pseudo bond graphs are

mainly used in process engineering systems modeling. The systematic construction and the use of a bond graph model remain applicable, but a pseudo bond graph cannot be connected to a real bond graph model.

3 The Bond Graph Elements

Similar behavior, from an energy processing point of view, leads to some *generic elements*. Each of them is governed by a *generic constitutive law* and represents a group of real physical elements. For example, a generic element, called a *capacitor*, has the following corresponding real elements: a linear spring in the mechanical translation field, a torsion spring in the mechanical rotation field, a capacitor in the electrical energy field, a storage tank in the hydraulic energy field and a heated enclosure in the thermal energy field. No matter the system type, the bond graph modeling method uses only nine generic elements for constructing its model.

3.1 Sources of Power

The elements providing power are named sources and are denoted by S . A source imposes one of the power variables (the *cause* physical quantity) and the other will result from the system necessities (the *effect* physical quantity). If the source imposes the effort, then it is called an *effort source*, and is denoted by S_e . If the source imposes the flow, then it is called a *flow source*, and is denoted by S_f . When the cause of the physical quantity must be modified, according to the system behavior via a feedback control, the source is called a *modulated source* and capital M is added to the notation (MS_e or MS_f). Power consumption for doing a work can be modeled by a source and, in these circumstances, the source is called a *sink* and is considered a *negative source*. An intuitive example is a weight raised by a lifting system.

3.2 Storage Elements

The elements that store energy in an *inductive* or *inertial* manner are denoted by I and are called *inductances* or *inertial* elements. The generic constitutive equation is $p - k_I f = 0$, where k_I is the parameter of I element. The elements that store energy in a *capacitive* manner are denoted by C and are called *capacitors* or *capacitive* elements. The generic constitutive equation is $k_C e - q = 0$, where k_C is the parameter of C element. In Table 3, we present the I and C elements in the principal energy fields.

Table 3 I, C and R elements in the main energy fields

Energetic field	I element parameter constitutive equation	C element parameter constitutive equation	R element parameter constitutive equation
Mechanical translation field	Body in translation motion $k_I = m$ [kg]-mass $p - mv = 0$	Linear spring $k_C = 1/k_e$, k_e [N/m]-spring stiffness $F - k_e x = 0$	Linear damper $k_R = \gamma$ [Ns/m]-viscous friction coefficient $F - \gamma v = 0$
Mechanical rotation field	Body in rotation motion $k_I = J$ [kgm ²]-moment of inertia $L - J\omega = 0$	Torsion spring $k_C = 1/k_t$, k_t [Nm]-torsion spring stiffness $M - k_t \omega = 0$	Torsion damper $k_R = \gamma_t$ [Nms]-torsion viscous friction coefficient $M - \gamma_t \omega = 0$
Electrical field	Coil $k_I = L$ [H]-inductance $\Phi - Li = 0$	Capacitor $k_C = C_e$ [C]-capacity $C_e u - q = 0$	Resistance $k_R = R_e$ [Ω]-electric resistance $u - R_e i = 0$
Hydraulic field	Fluid in laminar flow $k_I = L_f$ [kg/m ⁴]-fluid inductance $p_P - L_f Q = 0$	Tank $k_C = C_f$ [m ⁴ s ² /kg]-fluid capacity $V - C_f P = 0$	Pipe of constant area $k_R = R_f$ [kg/s/m ⁴]-fluid resistance $P - R_f Q = 0$
Thermal field	–	Heated substance mass $k_C = C_t$ [J/kg]-thermal capacity $Q - C_t T = 0$	Wall crossed by a heat flux $k_R = R_t$ [K/W]-thermal resistance $T - R_t \dot{Q} = 0$

3.3 Dissipative Elements

There are elements that irreversibly transform energy into heat and dissipate heat in the environment. These elements are called *resistors*, *dissipators* or *resistive elements* and are denoted by R . The generic constitutive equation is $e - k_R f = 0$, where k_R is the parameter of the R element, whose elements in the principal energy fields are presented in Table 3.

3.4 Transformers and Gytrators

There are two conserving power bond graph elements, which transform the power components, according to the following equation:

$$e_{in} f_{in} = e_{out} f_{out}. \quad (2)$$

The first element is called a transformer and is denoted by TF . It has a transformation ratio k_{TF} , which permits us to write two equations that connect efforts

with efforts and flows with flows, as shown in Fig. 9. Some examples are: the gear, the pulley, the lever and the electrical transformer.

The second element is called a gyrator and is denoted by GY . It has a transformation ratio k_{GY} , which permits us to write two equations that connect efforts with flows, as shown in Fig. 9.

3.5 Connection Elements

The constitutive elements of any system are connected to each other and the power is transmitted between them. The connection between the bond graph elements is made by two kinds of elements, named *one junction*, denoted by $J1$ or simply 1 and *zero junction*, denoted by $J0$ or simply 0. Both elements are of power conservative type. On April 24, 1959, when Henry Painter invented them, he understood that the bond graph methodology is “complete and constituted a formal discipline”.

The one junction (1) connects elements having the same flow variable and the zero junction (0) connects elements having the same effort variable. The powers getting in the junction have a plus sign and the powers getting out the junction have a minus sign. The principles associated with these junctions may be regarded as generalizations of the well-known Kirchhoff’s laws.

When n elements are connected at a one junction, the power conserving equation has the form:

$$\sum_{j=1}^n \alpha_j P_j = \sum_{j=1}^n \alpha_j e_j f_j = 0, \quad \alpha_j \in \{-1, +1\} \quad j = \overline{1, n}. \quad (3)$$

Taking into account that the flows are equal, the one junction constitutive relationships are:

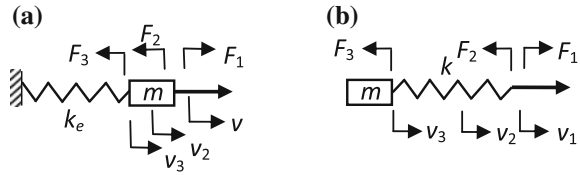
$$f_1 = f_2 = \dots = f_n \text{ and } \sum_{j=1}^n \alpha_j e_j = 0, \quad \alpha_j \in \{-1, +1\} \quad j = \overline{1, n}. \quad (4)$$

For a zero junction, Eq. 3 stands, but in this case, the efforts being equal, the constitutive equations become:

$$e_1 = e_2 = \dots = e_n \text{ and } \sum_{j=1}^n \alpha_j f_j = 0, \quad \alpha_j \in \{-1, +1\} \quad j = \overline{1, n}. \quad (5)$$

Two systems, composed of three elements, namely: an inertia element (the mass m), an effort source (the force F_1) and a capacitor element (the linear spring of stiffness k_e) are represented in Fig. 1. The power of force F_1 is positive and the powers of the force of inertia F_2 and of the elastic force F_3 are negative.

Fig. 1 Example of three elements connected to: **a** one junction **b** zero junction



In Fig. 1a, the three elements have the same velocity and they are connected to a one junction. The constitutive equations are:

$$v_1 = v_2 = v_3 \text{ and } F_1 - F_2 - F_3 = 0. \tag{6}$$

In Fig. 1b, the elements are acted upon by the same force and they are connected to a zero junction. The constitutive equations are:

$$F_1 = F_2 = F_3 \text{ and } v_1 - v_2 - v_3 = 0. \tag{7}$$

4 Causality in Bond Graphs

The causality issue is essential in bond graph modeling and its elements are shown below.

4.1 Bonds and Ports

The graphical symbol of the power exchange between the bond graph elements must have the possibility to emphasize, on one hand, a simultaneous transmission of the two components of the power and, on the other hand, the direction of the power exchange. A half of an arrow, named a *bond*, is chosen to do this, as it is shown in Fig. 2.

The power is provided by the generic bond graph element *A* and is received by the generic bond graph element *B*. The flow is usually written on the half arrow side.

Another important problem is to graphically emphasize which of the power signals is *cause* (*input*) and which is *effect* (*output*) for the elements *A* and *B*. We can have two situations for the bond presented in Fig. 2, as shown in Fig. 3.

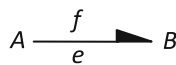
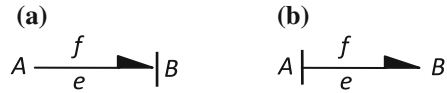


Fig. 2 The graphical representation of the power transmission between two generic bond graph elements *A* and *B* by a bond

Fig. 3 The two possible causal aspects for the bond presented in Fig. 2



Fig. 4 The two causal augmented bond for the situations presented in Fig. 3



In Fig. 3a, the element A has the flow as cause and the effort as effect. The element B has the effort as cause and the flow as effect. In Fig. 3b, the situation is reversed.

In order to graphically express these situations, a causal stroke perpendicular to the bond is drawn at the end where the element having the effort as cause is located. The causal bond for the situation depicted in Fig. 3a is represented in Fig. 4a, while for the situation depicted in Fig. 3b, the causal bond is represented in Fig. 4b. A bond provided with a causal stroke is called a *causal augmented bond*. There is no connection between the direction of the half arrow and the position of the causal stroke.

A point where a bond graph element exchanges power with another is named a *port*. There are elements with *one port* (sources, dissipative elements, inertia elements and capacitive elements), elements with *two ports* (transformers and gyrators) and *multiport* or *n-port* elements (junctions).

4.2 Causality Assignment

The causality assignment to the bonds is related to the types of the bond graph elements which are connected and this assignment should respect some rules. The parameter of the element is written near the symbol of the element and is separated by a colon.

The sources have a unique possibility for causal assignment, as illustrated in Fig. 5. The case of negative sources is also presented.

I and C elements can have two kinds of causality. When the input must be integrated in order to obtain the output, the element has *integral causality*, as is shown in Figs. 6a and 7b.

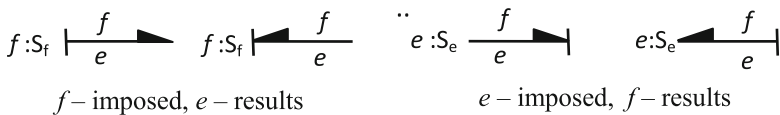


Fig. 5 Causality of the sources

$$\begin{array}{cc}
 \text{(a)} & \text{(b)} \\
 k_I:I \left| \begin{array}{c} \blacktriangle f \\ \hline e \end{array} \right. & f(t) = \frac{1}{k_I} \int_{t_0}^t e(\tau) d\tau + f(t_0) & k_I:I \left| \begin{array}{c} \blacktriangle f \\ \hline e \end{array} \right. & e(t) = k_I \frac{df}{dt}
 \end{array}$$

Fig. 6 Causality of I elements: **a** integral causality **b** derivative causality

$$\begin{array}{cc}
 \text{(a)} & \text{(b)} \\
 k_C:C \left| \begin{array}{c} \blacktriangle f \\ \hline e \end{array} \right. & f(t) = k_C \frac{de}{dt} & k_C:C \left| \begin{array}{c} \blacktriangle f \\ \hline e \end{array} \right. & e(t) = \frac{1}{k_C} \int_{t_0}^t f(\tau) d\tau + e(t_0)
 \end{array}$$

Fig. 7 Causality of the C elements: **a** derivative causality **b** integral causality

$$\begin{array}{cc}
 \text{(a)} & \text{(b)} \\
 k_R:R \left| \begin{array}{c} \blacktriangle f \\ \hline e \end{array} \right. & e(t) = k_R f(t) & k_R:R \left| \begin{array}{c} \blacktriangle f \\ \hline e \end{array} \right. & f(t) = \frac{1}{k_R} e(t).
 \end{array}$$

Fig. 8 Causality of R elements: **a** resistance causality **b** conductance causality

When the input must be differentiated in order to obtain the output, the elements I and C have *derivative causality*, as is shown in Figs. 6b and 7a. The half arrows point always to I and C elements, because they accumulate energy. The integral causality is always preferred to derivative causality.

R elements can have *resistance causality* if the input is the flow and *conductance causality* if the input is the effort, as can be seen in Fig. 8a, b, respectively. The half arrows point always to a R element.

Each of TF and GY elements can have two kinds of causality, as shown in Fig. 9, where the constitutive equations are written on the right-hand side. The half arrows must be oriented in the same direction, imposed by the power flow.

$$\begin{array}{ccc}
 \begin{array}{c} \frac{f_1}{e_1} \blacktriangle \text{TF} \frac{f_2}{e_2} \blacktriangle \\ \dots \\ k_{TF} \end{array} & \begin{array}{l} e_2 = k_{TF} e_1 \\ f_1 = k_{TF} f_2 \end{array} & \begin{array}{c} \frac{f_1}{e_1} \blacktriangle \text{TF} \frac{f_2}{e_2} \blacktriangle \\ \dots \\ k_{TF} \end{array} & \begin{array}{l} e_1 = \frac{1}{k_{TF}} e_2 \\ f_2 = \frac{1}{k_{TF}} f_1 \end{array} \\
 \begin{array}{c} \frac{f_1}{e_1} \blacktriangle \text{GY} \frac{f_2}{e_2} \blacktriangle \\ \dots \\ k_{GY} \end{array} & \begin{array}{l} e_2 = k_{GY} f_1 \\ e_1 = k_{GY} f_2 \end{array} & \begin{array}{c} \frac{f_1}{e_1} \blacktriangle \text{GY} \frac{f_2}{e_2} \blacktriangle \\ \dots \\ k_{GY} \end{array} & \begin{array}{l} f_1 = \frac{1}{k_{GY}} e_2 \\ f_2 = \frac{1}{k_{GY}} e_1 \end{array}
 \end{array}$$

Fig. 9 Causality of TF and GY elements

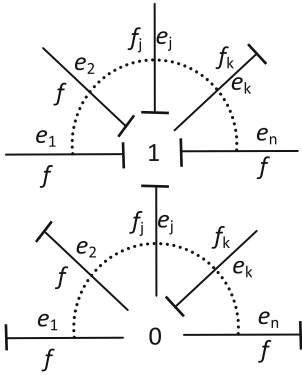


Fig. 10 Causality of J1 and J0 elements

The causality of J1 and J0 elements is imposed by one bond, as shown in Fig. 10. The half arrows are not represented, because they have no connection with causality assignment.

5 Construction of the Bond Graph Model

There is a specific methodology for constructing the bond graph model of mechanical systems and another one for non-mechanical systems.

5.1 Construction of Acausal Bond Graph Models for Mechanical Systems

The construction of an acausal bond graph model for mechanical systems involves the following steps:

1. Consider a J1 element for each well individualized absolute velocity in the system, that is for each I element.
2. Add a J0 element for each velocity resulting from an algebraic sum of two velocities coming from two J1 elements that is, for each relative velocity.
3. Attach the S_e , S_f , I, C and R elements to the corresponding J1 and J0 elements.
4. Introduce TF and GY elements between the corresponding J1 elements.
5. Assign the bonds with half arrows according to the flux of power between the elements.
6. Simplify the bond graph, if it is needed, and then, put numbers on the bonds.

$$f_j = f_k \quad j = \overline{1, n} \quad j \neq k$$

$$e_k = -\frac{1}{\alpha_k} \sum_{\substack{j=1 \\ j \neq k}}^n \alpha_j e_j \quad \alpha_j, \alpha_k \in \{-1, +1\}$$

$$e_j = e_k \quad j = \overline{1, n} \quad j \neq k$$

$$f_k = -\frac{1}{\alpha_k} \sum_{\substack{j=1 \\ j \neq k}}^n \alpha_j f_j \quad \alpha_j, \alpha_k \in \{-1, +1\}.$$

5.2 Construction of Acausal Bond Graph Models for Non-mechanical Systems

The construction of an acausal bond graph model for non-mechanical systems involves the following steps:

1. Consider a J0 element for each point characterized by a well—individualized potential, pressure or temperature.
2. Consider a J1 element for each S_e , S_f , I, C and R element. J1 element is introduced between two J0 elements that represent the potential, the pressure or the temperature, elements S_e , S_f , I, C and R are connected to.
3. Introduce TF and GY elements.
4. Assign the bonds with half arrows, according to the power flow direction between the elements.
5. Identify the elements J0, characterized by a reference (usually null) potential, pressure or temperature, for example the mass of an electric circuit, the atmospheric pressure or the reference temperature of the system (e.g. 0 °C, 20 °C or 0 K). Such a J0 element is eliminated with all its adjacent bonds.
6. Simplify the bond graph, if it is needed, and then, put numbers on the bonds.

5.3 Construction of the Causal Bond Graph

The bond graph model is finished after the causal assignment of bonds. This operation is fulfilled by following the next three steps:

Step 1

- 1.1. A source is chosen and is causally assigned, getting the imposed causality of the element.
- 1.2. A unique causality assignment results for some adjacent bonds, because of the specific rules for elements J0, J1, TF and GY. In this way, the causality propagates at all the bonds that can get an unambiguous causality.
- 1.3. Steps 1.1 and 1.2 are then applied to all sources.

Step 2

- 2.1. An I or C element is chosen and is causally assigned with integral causality.
- 2.2. Analogue operations to those specified in 1.2 are applied.
- 2.3. Steps 2.1 and 2.2 are then applied to all I and C elements.

Step 3

- 3.1. An unassigned R element is conveniently causally assigned.
- 3.2. Analogue operations to those specified in 1.2 are applied.
- 3.3. Steps 3.1 and 3.2 are then applied to all R elements.

In some cases, it is not possible to have integral causality for all I and C elements, without violating the causality assignment rules for one or more J1, J0, TF and GY elements. In these cases one or more energy storage elements will be in derivative causality.

6 Exploitation of Bond Graph as a Meta-Model

The bond graph model gives rise to three ways for using it. The first way is to deduce a *differential or algebraic-differential system of equations*, based on the bond graph model. The second way is to obtain another graphical model, named a *block diagram model*, which can be very easily implemented in appropriate software, such as MATLAB SIMULINK. The third way is to draw the bond graph model, as it was obtained, directly into *dedicated software*, such as 20sim or many others. The last two ways do not require solving any equation.

6.1 Deriving State-Space Models

The mathematical model is a state-space model in energy variables, when the bond graph model *does not contain* storage elements in *derivative causality*.

When the constitutive equations of the elements are linear, the next steps should be followed:

1. Identify the variables $e(t)$ of the effort sources and $f(t)$ of the flow sources.
2. Identify the energy variables p_I and q_C and the power variables f_I and e_C of I and C elements.
3. Write the equations $\dot{p}_I(t) = e_I(t)$ and $\dot{q}_C(t) = f_C(t)$ for each I and C element.
4. Make explicit each $e_I(t)$ and $f_C(t)$ by using the constitutive equations of R, J0, J1, TF and GY elements. The state-space system of equations in terms of variables p_I and q_C is finally obtained, based on the constitutive equations of I and C elements: $f_I = \frac{1}{k_I} p_I$ and $e_C = \frac{1}{k_C} q_C$. The variables of sources will occur, as well.

The number of equations equals the number of storage elements. The system can be easily expressed in terms of power variables f_I and e_C .

The system of equations will contain algebraic equations if the bond graph model contains storage elements in derivative causality, because the energy variables of the storage elements in derivative causality are not independent. The mathematical model is a differential-algebraic system in this case.

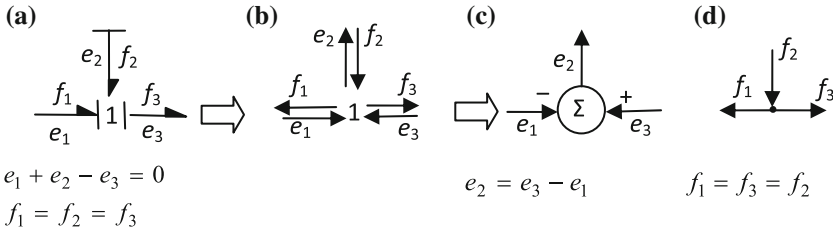


Fig. 11 Transformation of a J1 element from bond graph to block diagram model. **a** J1 element. **b** Graph of signal processing. **c** Summation. **d** Multiple output node

6.2 Deriving Block Diagram Models

The block diagram model is very easily obtained from the bond graph model, by simply transforming the bonds into two signals. One signal is for the effort and one signal is for the flow. The direction of the signals is given by the causality. The signal associated to effort is directed toward the element located near the causal stroke and the signal associated to flow is in the opposite direction. A graph called a *graph of signals processing* is obtained after this operation. This graph is then transformed into a block diagram model. Each J0 and J1 element is split in a *multiple output node-summation* pair. In the case of a J1 element, the summation is assigned to the efforts and the multiple output node is assigned to the flows. In the summation, all the efforts are inputs and only one effort is output. The output effort corresponds to the bond that imposes the velocity on each J1 element. Each effort gets a sign resulting from the equilibrium equation of the efforts. In the multiple output node, all the flows are output, except one, which is input. The input flow corresponds to the bond that imposes the velocity on J1 element. An example for three bonds connected to a J1 is given in Fig. 11.

The element J0 behaves similar to element J1, but the role of flow and effort are reversed.

The elements I and C in integral causality become *integrator blocks*, while I and C elements in derivative causality become *derivative blocks*. The elements R become *gain blocks*. The TF element is replaced by two gain blocks connecting effort to effort and flow to flow, respectively. The element GY is replaced by two gain blocks connecting efforts to flows. The sources are replaced by *source blocks*. A block diagram, processing only signals, finally results.

6.3 Software Environment for the Bond Graph Method

Dedicated software has been especially created for bond graph models. The bond graph model is directly drawn on the computer and, after introducing the numerical values of all required parameters, the simulation may run on its own. It is the most

amazing modality of exploiting the bond graph model. The usual software is presented in [2].

7 An Illustrative Example—Use of Bond Graph Method for a Hydro-Mechanical System

The bond graph modeling method is illustrated by an example consisting in a hydro-mechanical system, depicted in Fig. 12. The hydraulic part is composed of a constant pressure pump of magnitude $P = 1,000,000 \text{ N/m}^2$, which supplies oil to a hydraulic cylinder, through a pipe. The piston radius is $r = 0.06 \text{ m}$. The length of the pipe is $l = 1 \text{ m}$ and the area is $A = 0.002827 \text{ m}^2$. The dynamic viscosity of the hydraulic oil is $\mu = 0.1 \text{ Ns/m}^2$ and the density is $\rho = 900 \text{ kg/m}^3$. The hydraulic inertia of the oil in the pipe $L_h = \rho l/A = 318,309.88 \text{ kg/m}^4$ and the resistance of the pipe $R_h = 128 \mu l/A = 314,380.13 \text{ kg/(sm}^4)$, for a presumed laminar flow.

The mechanical part consists in a spring located on the other part of the piston, having stiffness $k_e = 50,000 \text{ N/m}$. The piston rod pushes an external weight of mass $m = 100 \text{ kg}$, including the mass of the piston and of its rod. The mass is subjected to the action of a damper with the proportionality coefficient $\gamma = 4000 \text{ Ns/m}$. The mass must overcome a constant force $F = 1000 \text{ N}$, as well.

The bond graph of the hydraulic part of the system is depicted in Fig. 13. There are three J0's for three absolute pressures. The first J0 is considered for the reference pressure p_0 , that is the atmospheric pressure, the second J0 is considered for the pressure p_1 of the source and the third J0 is considered for the pressure p_2 in the hydraulic piston. There are three J1 elements between them. The source of pressure is connected to the left one. The elements I and R, corresponding to the inertia of the hydraulic oil in the pipe and to the resistance of the pipe to oil flow, are connected to the middle one. A negative source of pressure (a sink), for the pressure exerted by the fluid on the piston, is connected to the right one. The final bond graph, obtained after removing the reference pressure together with its bonds, is represented in the right-hand side of Fig. 13.

The bond graph of the system mechanical part is depicted in Fig. 14. The piston is subjected to a force F_p , due to oil pressure that has been modeled by an effort source, and to an elastic force F_e , exerted by the spring, which has been modeled by a C element. The viscous friction force is modeled by an R element and the inertia

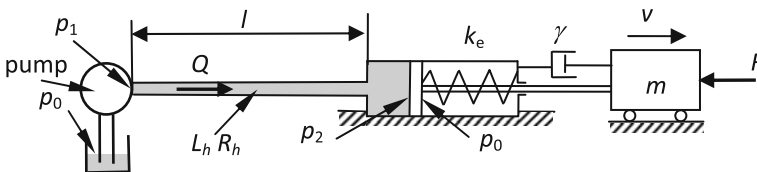


Fig. 12 The hydro-mechanical system

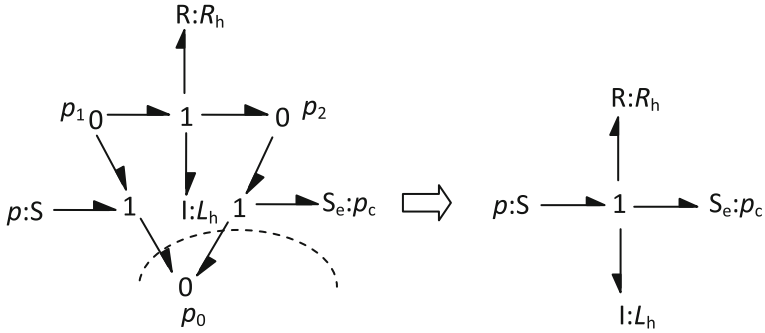


Fig. 13 The acausal bond graph model of the hydraulic part

of the mass m by an I element. The force F that must be overtaken is modeled by a negative source of effort. All these elements are connected to a J1 element corresponding to the common velocity of the mass, of the rod and of the piston.

The two bond graphs are connected by a TF element, based on the constitutive equations $F_p = p_c A$ and $Q = vA$, where A is the piston area, v is the piston velocity and Q is the volume flow.

The final bond graph is depicted in Fig. 15. The causality is assigned beginning with the two sources of effort. Then, the element I, corresponding to the mass m , is assigned in integral causality and all the bonds will receive a causal stroke. The notations for the elements and variables in connection with a bond will have as subscripts quite the bond number, for example: I_9 , e_1 or f_4 .

Fig. 14 The acausal bond graph model of the mechanical part

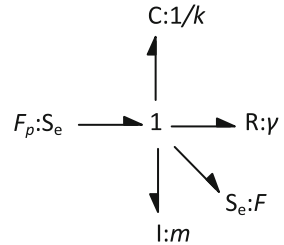
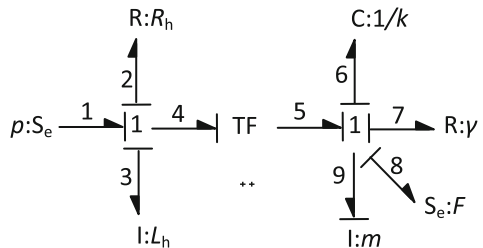


Fig. 15 The bond graph model of the whole system



Because the element I_3 is in derivative causality, the mathematical model consists of two differential equations and an algebraic one:

$$\dot{p}_8 = -\left(\frac{R_h A^2}{m} + \frac{\gamma}{m}\right)p_8 - \dot{p}_3 A - k_e q_6 - pA - F \tag{8}$$

$$\dot{q}_6 = \frac{1}{m} p_8 \tag{9}$$

$$p_3 = \frac{L_h A}{m} p_8. \tag{10}$$

This is not an unusual situation and it can be solved in two ways. The first one is to differentiate Eq. 10 and to consider a three differential equations system whose unknowns are: p_8 , q_6 and p_3 . The second one is to differentiate Eq. 10, to replace \dot{p}_3 in Eq. 8 and to finally obtain a system of two differential equations. In both situations, the systems must be expressed in an explicit form in order to perform a numerical simulation.

The bond graph shown in Fig. 15 generates the block diagram model pictured in Fig. 16. This is the most useful feature of bond graph modeling method, because any signal in the system can be seen and saved by simply attaching a scope. The derivative block du/dt must be replaced by a transfer function block $s/(as + 1)$,

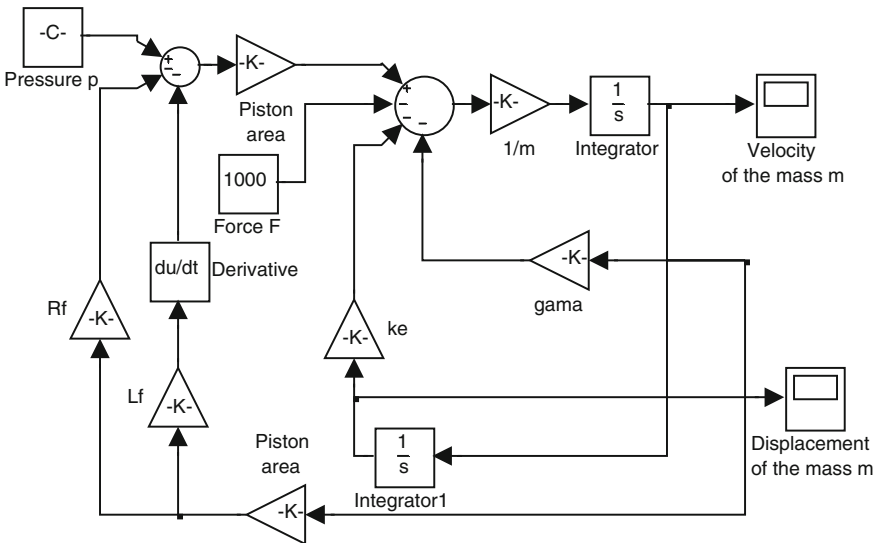


Fig. 16 The system block diagram model obtained from the bond graph model

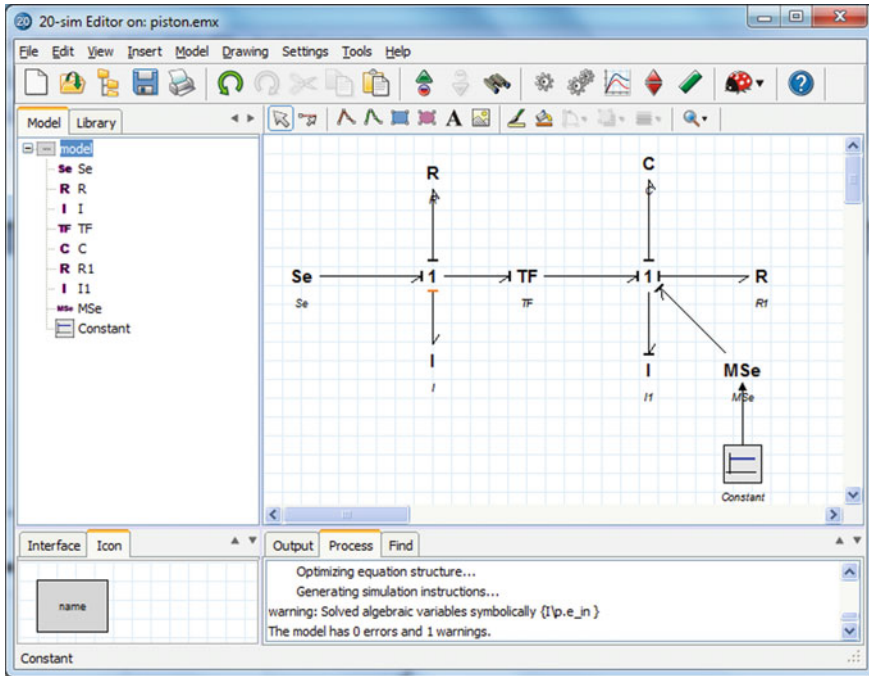


Fig. 17 The bond graph model in 20sim

which approximates the operation of differentiation [55]. This procedure prevents the occurrence of an error, for some sets of system parameter values. Possible errors are produced by the derivative block during the numerical simulation. The value of constant a , from the transfer function, must be estimated according to the frequency domain associated with the regular system operation, that is:

$$\frac{1}{a} \geq 10 \cdot (\max|\lambda_j|), \quad j = 1, 2, \dots, n \tag{11}$$

where: λ_j ($j = 1, 2, \dots, n$) are the eigenvalues of the system matrix. In this case, a very good value for a is 0.02.

The construction of the bond graph model in dedicated software is also a good option. In Fig. 17, the system bond graph created in 20sim is shown. This is an object oriented modeling method.

The diagram pictured in Fig. 18 shows the displacement of mass m . A complete analysis of system behavior, based on Eqs. 8, 9 and 10 is performed in [56].

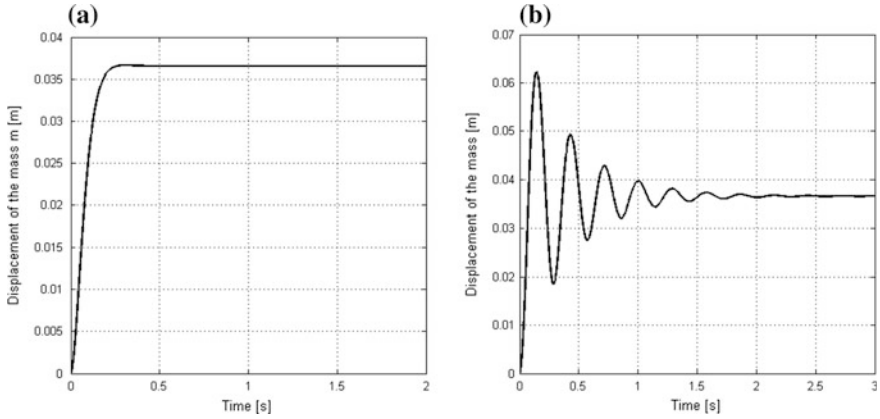


Fig. 18 The displacement of the mass m : **a** for $\gamma = 4000$ Ns/m **b** for $\gamma = 200$ Ns/m

8 Conclusions

The bond graph modeling method is based only on a deep and thorough analysis of the power flow in the system, from the source to the working elements. Any physical phenomenon misinterpretations are revealed by the violation of bond graph rules. So, the modeler has full control over the accurate interpretation of physical phenomena developed during system functioning.

There are three possibilities to exploit the bond graph model:

1. The first possibility consists in deducing a system of equations, usually a differential algebraic one. It has to be transformed in a system of differential equations after some algebraic manipulations. The system must be finally put in explicit form, in order to be numerically solved. These mathematical operations are encountered in any other modeling method.
2. The second possibility consists in getting another graphical model, named a block diagram. This model can be implemented in dedicated software, the most common being Matlab-Simulink, with all the benefits of such a model. The most important advantage consists in the possibility of visualizing any signal in the system, by simply attaching a scope to the desired signal. There is no need to solve any differential equation.
3. The third possibility consists in drawing the bond graph directly in specialized software, such as 20sim. The simulation can be performed after introducing the required parameters. There is no need to solve any differential equation, as well.

It is generally very useful in practice to apply two different modeling methods in case of very complex systems, in order to check the accuracy of results. The present paper strongly recommends the bond graph method to be one of these two.

The above issues confirm the main advantages of the bond graph method in system modelling.

References

1. Paynter, H.M.: Analysis and Design of Engineering Systems. M.I.T. Press (1961)
2. Borutzky, W.: Bond Graph Methodology. Springer, Development and Analysis of Multidisciplinary Dynamic System Models (2010)
3. Breedveld, P.-C., van Amerongen, J.: Dynamische systemen: modelvorming en simulatie met bondgrafen. Open Universiteit, Heerlen (1994)
4. Brown, F.T.: Engineering System Dynamics: A Unified Graph-Centered Approach, 2nd edn. CRC Press, New York (2006)
5. Dauphin-Tanguy, G.: Les Bond Graphs. Hermes Science Europe Ltd., Paris (2000)
6. Gawthrop, P.J., Smith, L.: Metamodelling: Bond Graphs and Dynamic Systems. Prentice Hall (1996)
7. Karnopp, D.C., Margolis, D.L., Rosenberg, R.-C.: System dynamics. Modeling and simulation of mechatronic systems, 4 edn. Wiley (2006)
8. Kypuros, J.A.: System Dynamics and Control with Bond Graph Modeling. CRC Press, New York (2013)
9. Mukherjee, A., Karmakar, R.: Modelling and Symulation of Engineering Systems through Bondgraphs. Narosa Publishing House, New Delhi (2000)
10. Mukhaerjee, A., Karmakar, R., Samantaray, A.K.: Bond Graph in Modeling, Simulation and Fault Identification. I.K. International Publishing House Pvt. Ltd., New Delhi (2006)
11. Păstrăvanu, O., Ibănescu, R.: Bond-Graph Language in Modeling and Simulation of Physical-Technical Systems. “Gheorghe Asachi” Publishing House, Iași (2001)
12. Samantaray, A.K., Ould Bouamama, B.: Model-based Process Supervision. A Bond Graph Approach. Springer, London (2008)
13. Singh, M.K., Singh, B.R., Faruqi, M.A.: Modelling and Simulation of Dynamic Half Car Using Bond Graph. LAP Lambert Academic Publishing, Saarbrücken, Germany (2014)
14. Taghouti, H., Mami, A., Jmal, S.: Nouvelle Technique de Modélisation et Simulation par Bond Graph. Applications aux Circuits Hauts Fréquences et Antennes Patch. Edition Universitaires Europeennes, France (2014)
15. Thoma, J.U.: Introduction to Bond Graphs and Their Applications. Pergamon Press, Oxford (1975)
16. Thoma, J.U.: Simulations by Bond Graphs-Introduction to a Graphical Method. Springer (1990)
17. Thoma, J.U., Ould Bouamama, B.: Modeling and Simulation in Thermal and Chemical Engineering (A Bond Graph Approach). Springer (2000)
18. Wang, D., Yu, M., Low, C.B., Arogeti, S.: Model-Based Health Monitoring of Hybrid System. Springer, New-York (2013)
19. Damić, V.: Modelling flexible body systems: a bond graph component model approach. Math. Comput. Model. Dyn. Syst. **12**(2–3), 175–187 (2006)
20. Das, S.: Mechatronic Modeling and Simulation Using Bond Graphs. CRC Press, New York (2009)
21. Muvengei, O., Kihui, J.: Using Bond Graphs in Simulating Hydro-Mechanical Systems: Case Study of an Excavator. VDM Verlag, Saarbrücken (2011)
22. Roddeck, W.: Grundprinzipien der Mechatronik. Modellbildung und Simulation mit Bondgraphen. Springer, Wiesbaden, Germany (2013)
23. Vijay, P., Samantaray, A.K., Mukherjee, A.: A bond graph model-based evaluation scheme to improve the dynamic performance of a solid oxide fuel cell. Mechatronics **19**, 489–502 (2009)
24. Sosnovsky, E., Forget, B.: Bond graph for spatial kinetics analysis of nuclear reactors. Ann. Nucl. Energy **56**, 208–226 (2013)
25. Sosnovsky, E., Forget, B.: Bond graph representation of nuclear reactor point kinetics and nearly incompressible thermal hydraulics. Ann. Nucl. Energy **68**, 15–29 (2014)
26. Mouhib, O., Jardin, A., Marquis-Favre, W., Bideaux, E., Thomasset, D.: Optimal control in bond graph formalism. Simul. Model. Pract. Theory **17**, 250–256 (2009)

27. Diaz-Zuccarini, V., LeFevre, J.: An energetically coherent lumped parameter model of the left ventricle specially developed for educational purposes. *Comput. Biol. Med.* **37**, 774–784 (2007)
28. Diaz-Zuccarini, V., Rafirou, D., LeFevre, J., Hose, D.-R., Lawfort, P.V.: Systemic modeling and computational physiology: the application of Bond Graph boundary conditions for 3D cardiovascular models. *Simul. Model. Pract. Theory* **17**, 125–136 (2009)
29. Le-Rolle, V., Hernandez, A., Richard, P., Buisson, J., Carrault, G.: A model of the cardioavascular system using Bond Graphs. *Innovation et Technologie en Biologie et Médecine-une Revue de Technologie Biomédicale ITBM-RBM* **26**, 243–246 (2005)
30. Narwal, A.-K., Vaz, A., Gupta, K.-D.: Study of dynamics of soft contact rolling using multiband graph approach. *Mech. Mach. Theory* **75**, 79–96 (2014)
31. Romero, G., Felez, J., Maroto, J., Mera, J.M.: Efficient simulation of mechanism kinematics using bond graphs. *Simul. Model. Pract. Theory* **17**, 293–308 (2009)
32. Xiaotian, L., Anlin, W.: Definitions of causality in bond graph model for efficient simulation mechanism. *Mech. Mach. Theory* **80**, 112–124 (2014)
33. Yutao, L., Di, T.: Dynamics modeling of planetary gear set considering meshing stiffness based bond graph. *Procedia Eng.* **24**, 850–855 (2011)
34. Selișteanu, D., Roman, M., Șendrescu, D.: Pseudo Bond Graph modeling and on-line estimation of unknown kinetics for a wastewater biodegradation process. *Simul. Model. Pract. Theory* **18**, 1297–1313 (2010)
35. Couenne, F., Jallut, C., Maschke, B., Breedveld, P.C., Tayakout, M.: Bond graph modeling for chemical reactors. *Math. Comput. Model. Dyn. Syst.* **12**(2–3), 159–174 (2006)
36. Nacusse, M.A., Junco, S.J.: Switchable structured bond: A bond graph device for modeling power coupling/decoupling of physical systems. *J. Comput. Sci.* **5**, 450–462 (2014)
37. Djeziri, M.A., Ould Bouamama, B., Merzouki, R.: Modelling and robust FDI of steam generator using uncertain bond graph model. *J. Process Control* **18**, 149–162 (2009)
38. Ragusila, V., Emami, M.-R.: Modelling of a robotic leg using bond graphs. *Simul. Model. Pract. Theory* **40**, 132–143 (2014)
39. Assimacopoulos, D.: Population interactions modeled by bond graphs. *Apply Math. Model.* **10**, 234–240 (1986)
40. Abbes, M., Farhat, A., Mami, A., Dauphin-Tanguy, G.: Pseudo bond graph model of coupled heat and mass transfer in a plastic tunnel greenhouse. *Simul. Model. Pract. Theory* **18**, 1327–1341 (2010)
41. Bakka, T., Karimi, H.R.: A bond graph approach to modeling and simulation of nonlinear wind turbine system. In: *Nonlinear Dynamics Science and Engineering*, vol. 3. Springer, Berlin, pp. 41–61 (2013)
42. Badoud, A.E., Khemliche, M.: *Bond Graph Modeling and Diagnosis in Wind Energy Conversion System*. LAP Lambert Academic Publishing, Saarbrücken (2014)
43. Kurniawan, A., Pedersen, E., Moan, T.: Bond graph modeling of wave energy conversion system with hydraulic power take-off. *Renew. Energy* **38**, 234–244 (2012)
44. Roman, M., Bobașu, E., Selișteanu, D.Ș.: Modelling of biomass combustion process. *Energy Procedia* **6**, 432–440 (2011)
45. Sanchez, R., Medina, A.: Wind turbine model simulation: A bond graph approach. *Model. Pract. Theory* **41**, 28–45 (2014)
46. Seo, K., Fan, Z., Hu, J., Goodman, E.-D.: Toward a unified and automated design methodology for multi-dynamic domain systems using bond graphs and genetic programming. *Mechatronics* **13**, 851–885 (2003)
47. Gawthrop, P.J., Ronco, E.: Estimation and control of mechatronic systems using sensitivity bond graphs. *Control Eng. Pract.* **8**, 1237–1248 (2000)
48. Kime, N.M., Ryan, M.J., Wilson, P.S.: A bond graph approach to modeling anuran vocal production system. *J. Acoust. Soc. Am.* **133**(6), 4133–4144 (2013)
49. Roy, S., Umanand, L.: Magnetic arm-switch-based three-phase series-shunt compensated quality AC power supply. *IET Electr. Power Appl.* **6**(2), 91–100 (2012)

50. Delgado, M., Profos, G.: Identifiability of Dynamic Systems Represented by Bond Graphs. *Math. Comput. Model. Dyn. Syst.* **5**(2), 89–112 (1999)
51. Chatty, N., Ould Bouamama, B., Gehin, A.L., Merzouki, R.: Signed bond graph for multiple faults diagnosis. *Eng. Appl. Artif. Intell.* **36**, 134–147 (2014)
52. Ibănescu, R.: Statics of Frames By Bond-Graphs. *Buletinul Institutului Politehnic Iași, Section: Mathematics. Theoret. Mech. Phys. L (LIV)*, **3–4**, 93–104 (2004)
53. Ibănescu, R., Ungureanu, C.: Approach of a Particle Statics Problem by Using the Bond-Graph Modeling Method. *Appl. Mech. Mater.* **657**, 599–603 (2014)
54. Ibănescu, R., Ibănescu, I., Melnici, R., Irimiciuc, N.: Similarity between sonic systems and electrical circuits emphasized by The Bond-Graph Method. *Buletinul Institutului Politehnic Iași, Section: Machine Construction, XLVII (LI)*, **3–4**, 387–393 (2001)
55. Ibănescu, R., Păstrăvanu, O.: Numerical problems in Block-Diagram Simulation of Bond-graph Models with Derivative Causality—A MATLAB-SIMULINK-based Case Study. *Buletinul Institutului Politehnic Iași, Section: Mathematics. Theoret. Mech. Phys. XLVIII (LII)*, **1–2**, 75–86 (2002)
56. Ibănescu, R.: The bond-graph model containing derivative causality. In: *Proceedings of 15th International Conference Modern Technologies, Quality and Innovation, Chișinău, Republic of Moldova* <http://www.modtech.ro/2011/papers.php>. pp. 493–496 (2011)

Structural Importance and Local Importance in Network Reliability

P. Tittmann and S. Kischnick

Abstract Network reliability analysis has interesting applications in areas such as computer and mobile networks. However, the computation of many important reliability measures (all-terminal reliability, reachability) turns out to be NP-hard. This statement applies to the computation of relevant reliability importance measures, too. In this paper we introduce *local importance measures* that describe the importance of an edge or vertex of the network in its local network neighborhood. Suitable scaling of the local neighborhood renders the computation of generally intractable reliability measures possible.

1 Introduction

The *importance* of an element x (vertex or edge) of a network (graph) G is a measure that describes the significance of the element for proper functionality. In order to make this idea more precise we assume that a given function $f : \mathcal{G} \rightarrow \mathbb{R}^+$ assigns a nonnegative real number $f(G)$ to any given graph G . This function is supposed to be monotone increasing with the performance (redundancy, reliability) of the network. Examples for those functions are edge and vertex connectivity, the number of vertex pairs that are reachable from each other, or the number of spanning trees of the graph. Let $G - x$ be the graph obtained from G by removal of element x (edge or vertex). Then

$$\frac{f(G) - f(G - x)}{f(G)}$$

P. Tittmann (✉) · S. Kischnick
Faculty of Mathematics, Sciences, and Computer Science,
University of Applied Sciences Mittweida, Technikumplatz 17, 09648 Mittweida, Germany
e-mail: peter@hs-mittweida.de

S. Kischnick
e-mail: kischnic@hs-mittweida.de

is a measure for the loss of performance when x has failed and hence also a measure for the importance of x . An importance measure is referred to as *structural importance* if it is solely dependent on the graph but not on availabilities of edges or vertices.

The concept of reliability importance was introduced by Birnbaum, [4]. Alternative structural reliability importance measures are presented in [12, 15, 16, 17]. Reliability importance measures have been applied for general binary and multi-state systems, see [14]. A game theoretic approach to reliability importance is presented in [8]. The interrelation of the importance of two different components has been investigated in [2] and [6]. For a more detailed introduction to system reliability and reliability importance, see [9, 19].

Unfortunately, it is shown that the computation of many reliability importance measures that are essential to systems and network analysis is **NP**-hard. A classical example is the Birnbaum importance given by

$$I_B(G, e) = \frac{\partial R(G)}{\partial p_e} = R(G/e) - R(G - e) \quad (1)$$

where $R(G)$ denotes the all-terminal reliability of a graph $G = (V, E)$, whose edges fail independently with given probability $p_e, e \in E$. As the computation of the all-terminal reliability is **NP**-hard (in fact it belongs to the class **#P**-complete, [18]) the computation of the Birnbaum importance is **NP**-hard, too. Also many structural importance measures that are based on counting all path sets or all cut sets of a graph are computationally intractable.

2 Structural Importance Measures

Structural importance measures provide an essential tool to rank the importance of components (edges or vertices) of a network in case that there are no reliability data available for edges and vertices of the network.

2.1 The Structural Birnbaum Importance

The measure of structural importance considered here is very likely the first measure that appeared in the literature; it was introduced by Birnbaum in [4]. Let $G = (V, E)$ be an undirected graph with m edges. We define a function $\Psi(G)$ by

$$\Psi(G) = \begin{cases} 1 & \text{if } G \text{ is connected,} \\ 0 & \text{if } G \text{ is not connected.} \end{cases} \quad (2)$$

If $F \subseteq E$ then we write $\Psi(V, F)$ instead of $\Psi((V, F))$. We consider a subgraph (V, F) of G operating if and only if (V, F) is connected. Consequently, a *path set* is in this context an edge subset F such that $\Psi(V, F) = 1$. Now let $e \in E$ be a fixed edge of G . An edge set $F \subseteq E$ is critical with respect to the edge e if

$$\Psi(V, F \cup \{e\}) - \Psi(V, F \setminus \{e\}) = 1. \quad (3)$$

Conversely, given a subset $F \subseteq E$ of edges, an edge $e \in E \setminus F$ is *essential* with respect to F if Eq. (3) is satisfied. The *structural Birnbaum importance* of an edge e of G is defined by

$$I_b(G, e) = \frac{1}{2^m} \sum_{F \subseteq E} [\Psi(V, F \cup \{e\}) - \Psi(V, F \setminus \{e\})]. \quad (4)$$

Thus the structural Birnbaum importance $I_b(G, e)$ counts critical sets of G with respect to e . We define

$$I_1(G, e) = \frac{1}{2^{m-1}} \sum_{F: e \in F \subseteq E} [\Psi(V, F) - \Psi(V, F \setminus \{e\})], \quad (5)$$

$$I_0(G, e) = \frac{1}{2^{m-1}} \sum_{F \subseteq E \setminus \{e\}} [\Psi(V, F \cup \{e\}) - \Psi(V, F)]. \quad (6)$$

Then we obtain by splitting the sum in Eq. (4)

$$I_b(G, e) = \frac{1}{2} (I_1(G, e) + I_0(G, e)). \quad (7)$$

A closer look at the Eqs. (5) and (6) shows that

$$I_1(G, e) = I_0(G, e) \quad (8)$$

as both sums count the same critical sets. This also shows that we can define the structural Birnbaum importance by Eq. (5).

2.1.1 Connected Spanning Subgraphs and the Tutte Polynomial

Let $G = (V, E)$ be an undirected graph. The graphs obtained from G by removal and contraction of an edge e are denoted by $G - e$ and G/e , respectively. Let $\tau(G)$ be a number of connected spanning subgraphs of G . Then we can easily conclude from Eq. (4) that

$$I_b(G, e) = \frac{\tau(G/e) - \tau(G - e)}{2^m}. \quad (9)$$

We call an edge of G that is neither a loop nor a bridge of G a *link*. The function $\tau(G)$ can be recursively computed by

$$\tau(G) = \begin{cases} 0 & \text{if } G \text{ is disconnected,} \\ 1 & \text{if } G \text{ is a tree,} \\ 2\tau(G - e) & \text{if } e \text{ is a loop,} \\ \tau(G/e) & \text{if } e \text{ is a bridge,} \\ \tau(G - e) + \tau(G/e) & \text{if } e \text{ is a link.} \end{cases} \quad (10)$$

The last representation follows from the well-known fact that $\tau(G)$ equals the evaluation $T(G; 1, 2)$ of the Tutte polynomial of G , see [5].

Theorem 1 *The computation of the structural Birnbaum importance is #P-hard.*

Proof It has been shown in [21] that the evaluation of $T(G; 1, 2)$ belongs to the class of #P-hard problems. Now let $G = (V, E)$ be a given graph and $v \in V$ an arbitrarily chosen vertex of G . We construct the new graph $G' = (V', E')$ by inserting a new vertex u and an edge $e = \{u, v\}$ in G . Then clearly a subset $F \subseteq E'$ is critical for e in G' if and only if (V, F) is a spanning subgraph of G . Hence we obtain $\tau(G) = 2^{|E'|} I_b(G', e)$.

Figure 1 shows an example graph with edges weighted by 100 times the structural Birnbaum importance. We see, as expected, that the bridges in the graph are the edges with maximum importance.

2.1.2 Modified Structural Birnbaum Importance

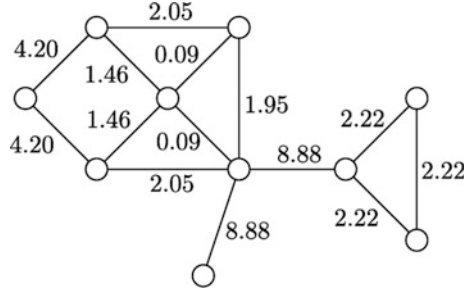
The normalizing factor, 2^{-m} , in Eq. (4) accounts for the powerset of E , which implies that in large sparse networks $I_b(G, e) \ll 1$ for all edges. The modified structural Birnbaum importance, defined by

$$I'_b(G, e) = \frac{\tau(G) - \tau(G - e)}{\tau(G)}, \quad (11)$$

provides a measure of structural importance with the property $I'_b(G, e) = 1$ for any bridge e of G . As $I'_b(G, \cdot)$ is a constant multiple of $I_b(G, \cdot)$ the importance ranking of edges is the same for both measures.

The reliability polynomial of a graph $G = (V, E)$ can be represented by

Fig. 1 Structural Birnbaum importance



$$R(G, p) = (1 - p)^m \sum_{F \subseteq E} \Psi(V, F) \left(\frac{p}{1 - p} \right)^{|F|}. \quad (12)$$

see, for instance, [3]. Consequently, $2^m R(G, 1/2)$ gives the number of connected spanning subgraphs of G , which implies

$$I_b(G, e) = R\left(G/e, \frac{1}{2}\right) - R\left(G - e, \frac{1}{2}\right) = \left. \frac{\partial R(G, p)}{\partial p_e} \right|_{p=\frac{1}{2}}, \quad (13)$$

which is the evaluation of the Birnbaum importance, given in Eq. (1) at $p = \frac{1}{2}$.

2.2 Spanning Trees and Electrical Resistance

The practical consequence of Theorem 1 is that the calculation of structural importance in large networks is impossible within reasonable time. There are several approaches to overcome this problem:

- The problem might be solvable in polynomial time when restricted to special graph classes. In fact it can be shown, see [1], that $I_b(G, e)$ can be efficiently calculated in graphs of bounded tree-width.
- Often polynomial-time algorithms can provide lower and upper bounds for the structural importance.
- An estimation for the desired measure can be obtained by Monte-Carlo simulation.
- In some cases another importance measure that is efficiently computable provides (almost) the same information.
- The desired importance measure can be *locally* calculated, i.e. with respect to a part of the network that is close to the edge (or vertex) to be investigated.

We will focus in the following on the latter two methods.

2.2.1 Spanning Trees

Let us consider a second importance measure in more detail. We choose the number $t(G)$ of spanning trees of a graph G as performance measure. A spanning tree is a minimal spanning subgraph of a given graph that ensures connectedness. Hence we may assume that a graph with many spanning trees is *more reliable* than a graph with a fewer number of spanning trees. This argument might be questionable when comparing completely different graphs. However, we can show that it is a suitable measure to study the effect of edge removal. We define

$$I_r(G, e) = \frac{t(G) - t(G - e)}{t(G)} = \frac{t(G/e)}{t(G)}. \quad (14)$$

The second equality results from the well-known decomposition formula $t(G) = t(G - e) + t(G/e)$, which is valid for any edge $e \in E(G)$. If e is a bridge of G then $G - e$ does not have any spanning trees, $t(G - e) = 0$, which implies $I_r(G, e) = 1$. Hence a bridge has maximum importance. Any edge of a series system (a tree) has importance 1. If G is a graph with two vertices that are linked with each other by m parallel edges then each edge has importance $1/m$. However, in both cases (series and parallel system), all edges have the same importance.

Equation (14) resembles the definition of structural Birnbaum importance by Eq. (6), whereas now the number of spanning trees of G is used to normalize the measure. The similarity of the two measures becomes even more obvious if we compare the two definitions given in Eqs. (14) and (11). The only difference is that we count in $I_r(G, e)$ only *minimum* connected spanning subgraphs. Hence we expect the importance measure I_r to provide a similar edge importance ranking than the one defined by I_b .

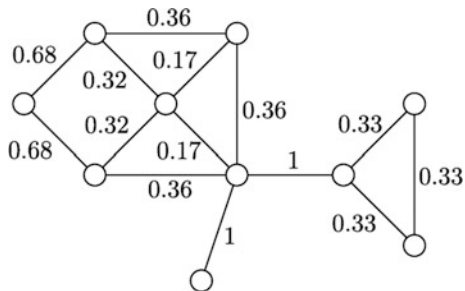
2.2.2 Electrical Resistance

There is a completely different interpretation of the fraction

$$\frac{t(G/e)}{t(G)}.$$

According to Kirchhoff laws, [11], this fraction is also the effective resistance that we measure between the endpoints u and v of the edge $e = \{u, v\}$ in the network $G = (V, E)$ where all the edges of E represent unit resistors, see also [5]. Hence we might call $I_r(G, e)$ the *resistance importance* of e . Interestingly, this new interpretation supports the understanding of $I_r(G, e)$ as an importance measure. The effective resistance between the end vertices u and v is 1 if and only if there is no other path than the edge e in G that connects u with v . On the other hand, if the resistance is small then there must be many other (short) paths that connect the end vertices of e making e less important for the connectivity of G . Figure 2 shows a

Fig. 2 A graph with edge weights $I_r(G, e)$



graph whose edges are weighted with $I_r(G, e)$. The total number of spanning trees of this graph is 198.

3 Flow and Distance Related Structural Importance Measures

We consider now a measure of importance that appeared first time in the context of social network analysis. The idea is that an edge is important if it is located on many shortest paths between different vertex pairs. A corresponding centrality measure has been introduced in [7] in order to determine the centrality of a vertex in a social network. An analog centrality measure with respect to the edges of a graph was employed in [10] in order to identify communities (dense subgraphs) in a graph. A nice overview about different kinds of centrality measures in social network analysis is given in [13].

3.1 Stress Centrality

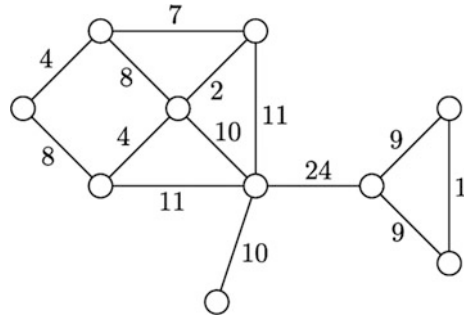
Let $G = (V, E)$ be a graph and $s, t \in V$. We denote by $\sigma_{st}(e)$ the number of shortest paths between s and t in G that contain the edge e and define

$$V^{(2)} = \{\{u, v\} | u, v \in V\}$$

as the set of all unordered pairs of vertices of V . The *stress centrality*, introduced in [20], of an edge e is defined by

$$c_s(e) = \sum_{\{s,t\} \in V^{(2)}} \sigma_{st}(e). \quad (15)$$

Fig. 3 Stress centrality as importance measure



Assume that traffic in a communication network is routed along shortest paths and the stress centrality $c_s(e)$ tries to evaluate the amount of flow that is routed through e . Hence we can consider the stress centrality of an edge as a measure of importance with respect to network reliability. Figure 3 shows the graph that has been also used in the preceding examples now weighted with the stress centralities of the edges. The stress centrality distinguishes the two bridges of the graph. Even more interesting, two non-bridge edges have a higher importance than one of the bridges.

3.2 Modified Stress Centrality

A slight modification of the stress centrality might yield a more appropriate importance measure for some applications. We define

$$\delta_{st}(e) = \begin{cases} 1 & \text{if } e \text{ is in a shortest } st\text{-path,} \\ 0 & \text{otherwise.} \end{cases}$$

The sum

$$I_p(G, e) = \sum_{\{s,t\} \in V^{(2)}} \delta_{st}(e) \tag{16}$$

gives the number of vertex pairs of G for which there exist a shortest path traversing the edge e .

3.3 *Betweenness Centrality*

We can refine the investigation of the importance of an edge by considering the possibilities of rerouting the flow in the network in case the edge fails. Suppose the flow between two vertices s and t is routed along a shortest path that traverses a given edge e . Then a failure of e is less important as long there are other paths of same length connecting s and t . The *betweenness centrality* takes this effect into account. It is defined by

$$c_b(e) = \sum_{\{s,t\} \in V^{(2)}} \frac{\sigma_{st}(e)}{\sigma_{st}}, \quad (17)$$

where σ_{st} denotes the number of shortest st -paths in G .

3.4 *An Importance Measure Based on Distances*

Another way to measure the importance of an edge in a graph is obtained by considering the effect of edge removal on distances between vertices in a graph. We denote by $d(u, v)$ the *distance* of two vertices u and v in a graph $G = (V, E)$, that is the length of shortest path between u and v in G . The distance $d(u, v)$ is defined to be infinite if there is no path between u and v . The *total distance* of a vertex v is

$$\text{td}(v) = \sum_{w \in V} d(v, w). \quad (18)$$

The *Wiener index* of G , defined by

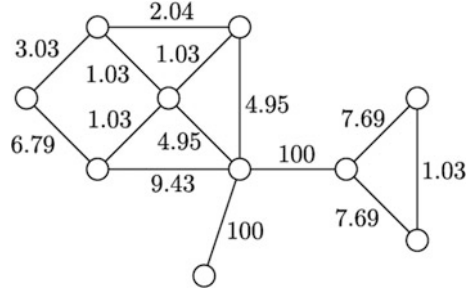
$$W(G) = \sum_{v \in V} \text{td}(v) = \sum_{\{u,v\} \subseteq V} d(u, v). \quad (19)$$

increases when an edge of G is removed. By

$$I_d(G, e) = 1 - \frac{W(G)}{W(G - e)} \quad (20)$$

we define the *distance importance* of the edge e in G . For communication networks, short paths between many vertex pairs are desirable. Hence the reciprocal of the Wiener index provides a suitable performance measure for networks, which explains the difference in Eq. (20). Figure 4 shows our example network this time with edges labeled according to Eq. (20), where $I_d(G, e)$ is multiplied with 100.

Fig. 4 A distance importance measure



4 Local Importance Measures

For large networks, the computation of structural importance measures might be time-consuming or even impossible. In order to obtain in this case at least good estimations of the importance of edges or vertices, we use the following approach. If an edge has a high importance in the network then it can be assumed that this edge also has a high importance in a certain *network neighborhood* of itself. Let $G = (V, E)$ be an undirected graph, $k \in \mathbb{N}$, and $v \in V$. The k -neighborhood of v in G is defined by

$$N_k(v) = \{w \in V \mid d(v, w) \leq k\}.$$

Consequently, the k -neighborhood of v consists of all vertices of G that have distance at most k from v . We define the k -neighborhood of an edge $e = \{u, v\}$ of G by

$$N_k(e) = N_k(u) \cup N_k(v).$$

Assume that $I(G, e)$ is any importance measure for the edge e of G . Then we define for any non-negative integer k the *local importance* $I(G, e, k)$ as

$$I(G, e, k) = I(G[N_k(e)], e).$$

Depending on the definition of the importance measure, $I(G, e, k)$ might be an approximation; it can also provide an upper or a lower bound for $I(G, e)$. Consider, as an example, the importance measure $I_r(G, e)$ defined in Eq. (14), which is the effective electrical resistance between the end vertices of e assuming all edges represent unit resistors. Building the neighborhood network $G[N_k(e)]$ means that we cut out a part of the network. This operation causes an increase of the resistance such that $I_r(G, e, k)$ is an upper bound for $I_r(G, e)$.

Table 1 shows the values of local importance measures for a grid graph of dimension 41×41 with respect to a central edge in the middle of the 21st column of the grid. Figure 5 shows the graph $G[N_3(e)]$, where G is a grid graph. The first column gives the parameter k that defines the local neighborhood of the edge. The

Table 1 Local importance measures

k	n	$I_r(G_s, e, k + 1)$	$I_r(G, e, k)$	$I'_p(G, e, k)$	$I_b(G, e, k)$
1	8	0.4597701	0.6000000	0.5000000	8.66667
2	18	0.4786778	0.5402299	0.4117647	28.3555
3	32	0.4868770	0.5213222	0.3709677	65.8332
4	50	0.4911341	0.5131230	0.3469388	127.106
5	72	0.4936175	0.5088659	0.3309859	218.235
6	98	0.4951893	0.5063825	0.3195876	345.278
7	128	0.4962458	0.5048107	0.3110236	514.274
8	162	0.4969896	0.5037542	0.3043478	731.248
9	200	0.4975327	0.5030104	0.2989950	1002.21
10	242	0.4979412	0.5024673	0.2946058	1333.18
11	288	0.4982561	0.5020588	0.2909408	1730.15
12	338	0.4985041	0.5017438	0.2878338	2199.12
13	392	0.4987027	0.5014959	0.2851662	2746.10
14	450	0.4988643	0.5012972	0.2828508	3377.08
15	512	0.4989975	0.5011357	0.2808219	4098.06

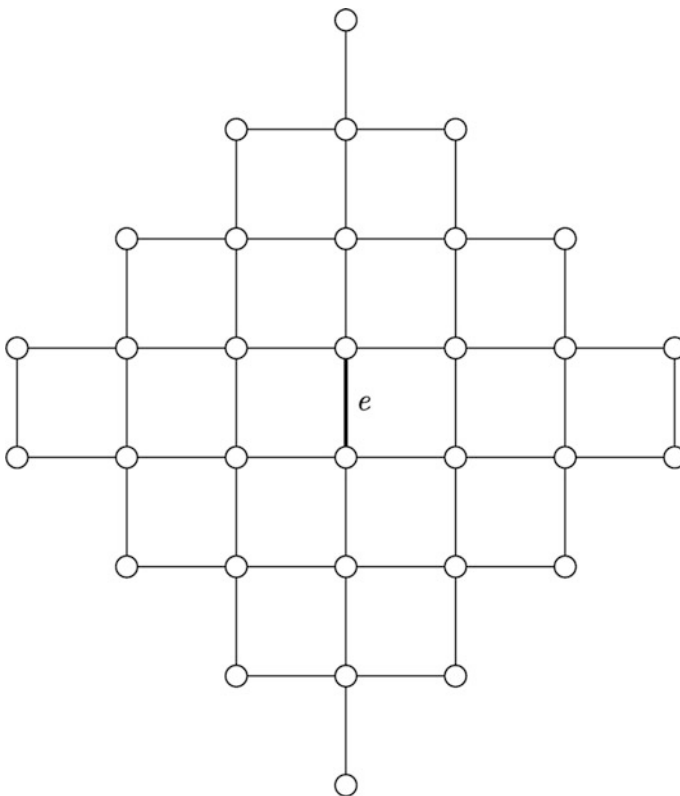


Fig. 5 The neighborhood graph of an edge in the grid for $k = 3$

second column shows the order of the neighborhood graph. The third column shows the resistance importance of $G_s[N_{k+1}(e)]$, which is the graph obtained from the edge neighborhood graph $G[N_{k+1}(e)]$ by merging all vertices that have distance $k+1$ from the edge e . This means that we shortcut all vertices of the “outer shell” of the neighborhood graph. The resulting value $I_r(G_s, e, k+1)$ is a lower bound for $I_r(G, e)$. The corresponding upper bound is given in the next column of Table 1. The values shown in column 5 of Table 1 present an importance measure obtained from I_p , see Eq. (16), by multiplication with $\frac{2}{n(n-1)}$, where n is the order of the neighborhood graph. This modification produces a normalized importance measure. The last column gives the betweenness centrality of the edge e in the respective neighborhood graph.

5 Summary and Conclusions

The structural importance of edges in graphs with respect to network reliability can be evaluated by a variety of local importance measures. Some of them, for instance the structural Birnbaum importance, are computationally intractable, whereas others, like the resistance importance, are computable in polynomially bounded time. An approach to overcome the computational hardness of the computation of importance measures is the introduction of local importance measures.

There remain, however, some interesting open questions:

- The importance ranking of edges depends on the selected structural importance measure. Which structural importance measures are closest to each other with respect to the ranking? Can we find bounds for the importance difference of edges?
- Numerical experiments suggests that the local importance quickly converges to the global structural importance when the neighborhood radius k grows. Is there a way to describe the quality of approximation in dependence on k ?

All structural importance measures considered here are defined with respect to edges of a graph. The given importance measures can be easily generalized for vertices of a graph or even for components of monotone binary systems.

References

1. Andrzejak, A.: An algorithm for the Tutte polynomials of graphs of bounded treewidth. *Discrete Math* **190**, 39–54 (1998)
2. Armstrong, M.J.: Joint reliability-importance of components. *IEEE Trans. Rel.* **44**, 408–412 (1995)
3. Beichelt, F., Tittmann, P.: *Reliability and maintenance: Networks and systems*. CRC Press (2012)

4. Birnbaum, Z. W.: On the importance of different components in a multicomponent system. In: Krishnaiah, P. R. (ed.) *Multivariate Analysis—II*, pp. 581–592. Academic Press (1969)
5. Bollobás, B.: *Modern Graph Theory*. Springer (1998)
6. Borgonovo, E.: The reliability importance of components and prime implicants in coherent and non-coherent systems including total-order interactions. *Eur. J. Oper. Res.* **204**, 485–495 (2010)
7. Freeman, L.C.: A set of measures of centrality based on betweenness. *Sociometry* **40**, 35–41 (1977)
8. Freixas, J., Puente, M.A.: Reliability importance measures of the components in a system based on semivalues and probabilistic values. *Ann. Oper. Res.* **109**, 331–342 (2002)
9. Gertsbakh, I., Shpungin, Y.: *Network Reliability and Resilience*. Springer (2011)
10. Girvan, M., Newman, M. E. J.: Community structure in social and biological networks. In: *Proceedings of the National Academy of Sciences*, vol. 99, pp. 7821–7826 (2002)
11. Kirchhoff, G.R.: Über die Auflösung der Gleichungen, auf welche man bei der Untersuchung der linearen Verteilung galvanischer Ströme geführt wird. *Ann. Phys. Chem.* **72**, 497–508 (1847)
12. Kochar, S., Mukerjee, H., Samaniego, F.J.: The signature of a coherent system and its application to comparisons among systems. *Naval Res. Logistics* **5**, 507–523 (1999)
13. Koschützki, D., Lehmann, K. A., Peeters, L., Richter, S., Tenfelde-Podehl, D., Zlotowski, O.: *Centrality Indices*. In: *Network analysis*. Springer (2005)
14. Levitin, G., Lisnianski, A.: Importance and sensitivity analysis of multi-state systems using the universal generating function method. *Reliab. Eng. Syst. Saf.* **65**, 271–282 (1999)
15. Meng, F.C.: On some structural importance of system components. *J. Data Sci* **7**, 277–283 (2009)
16. Myrvold, W.: Reliable network synthesis: Some recent developments. *Proceed. Eighth Quadrennial Int. Conf. Graph Theor, Combinatorics, Algorithms, Appl.* **II**, 650–660 (1999)
17. Navarro, J., Rychlik, T.: Reliability and expectation bounds for coherent systems with exchangeable components. *J. Multivar. Anal.* **98**, 102–113 (2007)
18. Provan, J.S., Ball, M.O.: The complexity of counting cuts and of computing the probability that a graph is connected. *SIAM J. Comput.* **4**, 777–788 (1983)
19. Rausand, M., Høyland, A.: *System reliability theory: Models, statistical methods, and applications*, 2nd edn. Wiley, Hoboken (2004)
20. Shimbel, A.: Structural parameters of communication networks. *Bull. Math. Biophys.* **4**, 501–507 (1953)
21. Welsh, D.J.A.: *Complexity: Knots. Cambridge University Press, Colorings and Counting* (1993)

Access Distribution Scheme to the Computer System Based on Fuzzy Logic

A. Shaikhanova, A. Zolotov, L. Dubchak, M. Karpinski
and V. Karpinskyi

Abstract This paper presents access distribution at computer system transmission of client-server data type. In order to enhance information protection fuzzy logic is applied to select the data encryption method. A server block diagram of such computer system has been developed based on the proposed fuzzy distribution of system access.

Keywords Client-server model · Fuzzy logic · Timing attack · Modular exponentiation · Resistance to side-channel attacks

1 Introduction

Client-server architecture is one of the most popular concepts in creating computer information systems. This architecture includes the following components:

A. Shaikhanova (✉) · A. Zolotov
Semey State University named after Shakarim, Semey, Kazakhstan
e-mail: igul7@mail.ru

A. Zolotov
e-mail: azol64@mail.ru

L. Dubchak
Ternopil National Economic University, Ternopil, Ukraine
e-mail: l_vasylkiv@rambler.ru

M. Karpinski
University of Bielsko-Biala, Bielsko-Biala, Poland
e-mail: mkarpinski@ath.bielsko.pl

V. Karpinskyi
The Techno Centre, Coventry University, Coventry, UK
e-mail: vkarpinskyi@gmail.com

- back-end (preservation and processing of information);
- client side (user’s instrument manual);
- network that provides interaction (exchange of information) between a client and a server.

Most of web-based systems are based on this architecture.

Such systems can be extremely varied and complex. The advantages of web-based systems based on client-server architecture are [4, 7]: minimum maintenance cost of business processes, maximum efficiency of data handling, easy maintenance, minimum cost of communication between business units, possibility of connection to the system from any computer with Internet access.

Each client of computer network is identified by its IP address and has its own “history” of using the information processing and transmission system regarding the presence of failures or data loss while transmitting ciphertext. This information is stored on a server that assigns to a user its level of access to information (Fig. 1).

However, when a violator performs the time attack or substitution of the IP-address stability of cryptosystem cannot be univocally provided [3, 4, 6].

Hundreds of thousands of network intrusions are registered during a year. However, taking under consideration that 80 % of computer crimes are not included in the official statistics because victims are afraid of publicity, which could undermine their trust of the partners and customers [12].

Modern information processing systems are represented as sophisticated software and hardware systems with specific information leakage channels accompanied with operational processing of information resources.

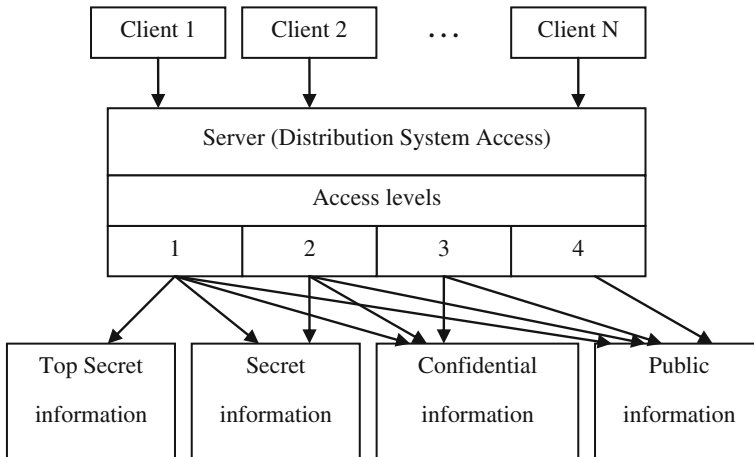


Fig. 1 Information access levels of computer system users

Thus, computer systems are exposed to a wide range of potential threats that causes necessity to anticipate a large list of functions and subsystems protection. First of all, it is necessary to protect the most informative channels of information leakage.

Overlapping problems of those channels are complicated by the fact that data protection procedures should not lead to a noticeable reduction in Computer System (CS) performance.

Since, access to information in most of CS is carried out remotely via the Internet; because of this it is essential to protect information against unauthorized access.

Threats to computer systems can be classified into nine features [4]:

1. For the purpose of threat implementation: breach of confidentiality; disturbing of the information integrity; disturbing of computer systems functioning.
2. According to the principle of the influence on the CS:
 - using a systems subject access (user, process) to the object (data file, data link, etc.);
 - using the covert channels (information transmission paths that allows for two cooperating processes to share information in a manner that disturbs the system security policy).

Interventions based on the first principle are simpler and more informative; it is easier to defend against them. Interventions based on the second principle are distinguished of level organization difficulty, lower information providing and difficulty in detection and removal.

3. The nature of the impact on CS:
 - active threat, that leads to a change of the system state and can be carried out either using the access or using both access and covered channels together;
 - passive threat, carried out by user observing of any side effects and their analysis. An example of passive impact may be a monitoring of the link between two network nodes. Passive impact does not lead to the system change. It is always associated with breach of confidentiality in the CS.
4. By reason of protection error use, this can be caused by one of the following:
 - inconsistency of security policy in real CS;
 - administrative management errors, which refers to incorrect implementation or support of security policies adopted by the CS;
 - errors in algorithms, relations between them and others that occur in the design phase of the program or set of programs. Therefore, they can be used in a wrong way, not as described in the documentation;
 - realization errors in algorithms (coding errors), links between them etc., which occur at the stage of implementation or debugging and can also be a source of undocumentation.

5. By manner of influence on the object of attack (in case of active impact):
 - direct impact on the object of attack (such actions are usually easily prevented with the help of access control);
 - impact on the system permission (including seizure of privileges);
 - mediate influence (through other users);
 - “masquerade”, in that case the user assigns itself different user authorities (pretends to be it);
 - “user blindfold”—when one user is forcing another to perform required actions, where the last may not suspect that; for this purpose virus can be used (it takes appropriate action and informs the one, who brought him, about the outcome).
6. By manner of influence on CS: in interactive mode and in batch mode.
7. By object of the attack:
 - CS in general (intrusion into the system), for that purpose usually is used method of “masquerade”, interception or fake password, “hacking” and access to CS through the network;
 - CS objects—data or programs, the system devices, data transmission channels;
 - CS subjects—users’ processes and sub processes, frequent case of such influence is loading of malicious virus to the environment of another process and its execution on behalf of that process;
 - data transmission channels—channels, data packets transmitted by communication channels, channel monitoring and traffic analysis (message flow, substitution or message modification in communication channels and in retranslation nodes, topology and characteristics changes of the network).
8. By used tools of attack (using either standard software or specially designed programs).
9. By object state of attack. The object of attack can be in one of three states:
 - saving—impact on an object is usually carried out by using the access;
 - transmission—effect provides either access to fragments of transmitted information or monitoring using covert channels;
 - processing—user process is the target object.

There are four standard methods that you can apply to restrict the access to information in a computer system [4, 12]:

- access control (the check of IP-address of each received packet, restriction of access with passwords, application of software tools);
- extension of password protection (response to a remote call, that is checking the password by a “recall”, continuous handshaking of connection—a system in which the server continually negotiates with the client computer during the connection session);

- encryption (the most popular asymmetric information security system is RSA, that enables creation of a sustainable digital signature);
- firewall (a combination of hardware and software to prevent access to information from the Internet).

These approaches do not provide a complete stability of a system since a violator can substitute an IP-address, intercept data packets that are transmitted through the communication channel and thus find out the password. Therefore to ensure the stability of a computer system that uses a client-server type network for data transfer it is necessary to consider the most dangerous attacks that can be carried out through the side channels of information leakage.

According to the ways of information interception, physical nature of data transmission channels, as well as the environment of information dissemination, the channels of leakage and interception of information can be divided into electromagnetic, electric, acoustic, cable local area networks (LANs), visual, inductive, parametric, bookmarks and viruses [8, 13].

Attacks of extraneous information leakage channels are characterized by less power than traditional attacks based on mathematical analysis of a cryptographic algorithm, but at the same time they are much more effective. Their research is presented in [2, 10]. The most dangerous attack of this type for the computer system is a timing attack [1], therefore development of the methods of counteraction to modern side channel attacks due to unauthorized information leakage is a relevant task.

2 RSA is a Modern System of Information Security

When developing a stable computer information security systems it is essential to provide restriction of the system functioning to bypass the subsystems protection and access delimitation. Cryptosystems based on elliptic curves have a high level of information security, but in practice, they are not widely used because of the implementation complexity [13].

The system of asymmetric cryptography allows realizing robust authentication of the parties, applying and verifying of digital signature, issuance and verification of public key certificates [9].

A security policy system should determine that authorised authority should issue a certificate of the public key only in the case that the key is given to an organization or an authorised person who will use this key. Thus specific organizational requirement should be developed.

Modern cryptosystem standard RSA algorithms usually use (name of the system is formed from the first letters of inventors' names—R. Rivest, A. Shamir, L. Adleman) with a key length of 1024 bits [8].

The RSA system that was first introduced in 1977 is perhaps the most popular system of protection of information with the public key.

The main operation that affects the stability and performance of asymmetric cryptosystems is modular exponentiation. The choice of modular exponentiation method that is resistant to time analysis and ensures advanced system performance is a priority task.

A lot of public-key cryptosystems use the function of discrete exponentiation [9]

$$f(n) = x^n \pmod{m} \quad (1)$$

where

n is an integer number ($1 \leq n \leq m - 1$),

m is a large number,

x is an integer number ($1 \leq x \leq m$).

To provide the sustainability of two-keys systems fairly large values of x and p must be used, therefore there is a need to use special methods to simplify and accelerate the calculation process of this function. Currently the most used methods are the following: binary method, β method, sliding window method, methods of fixed index, fixed basis methods and methods that use modules' special features [9].

A binary method uses binary image number $n = (n_{k-1} \dots n_0)_2$. This method is performed in two directions. When reading "left to right" x^n is written as [9]:

$$x^n = x^{(n_{k-1} \dots n_0)_2} = \left(\dots \left(\left((x^{n_{k-1}})^2 x^{n_{k-2}} \right)^2 \dots \right) x^{n_1} \right)^\beta x^{n_0}. \quad (2)$$

In the binary method based on read "right to left" the following record is used [9]:

$$x^n = x^{(n_{k-1} \dots n_0)_2} = \left(x^{2^0} \right)^{n_0} \left(x^{2^1} \right)^{n_1} \dots \left(x^{2^{k-1}} \right)^{n_{k-1}} = \prod_{\{i|n_i=1\}} x^{2^i}. \quad (3)$$

The B method is based on the image of exponent with the β base, that is $n = (n_{k-1} \dots n_0)_\beta$. This method is also performed in two ways. When reading "left to right" [9]:

$$x^n = x^{(n_{k-1} \dots n_0)_\beta} = \left(\dots \left(\left((x^{n_{k-1}})^\beta x^{n_{k-2}} \right)^\beta \dots \right) x^{n_1} \right)^\beta x^{n_0}. \quad (4)$$

When reading "right to left" [9]:

$$x^n = x^{(n_{k-1} \dots n_0)_\beta} = \left(x^{\beta^0} \right)^{n_0} \left(x^{\beta^1} \right)^{n_1} \dots \left(x^{\beta^{k-1}} \right)^{n_{k-1}} = \prod_{w=1}^{\beta-1} \left(\prod_{\{i|n_i=w\}} x^\beta \right)^w. \quad (5)$$

Sliding window method is based on an arbitrary partition into blocks (windows) of a binary image of exponent degree, that is $n = [w_{i-1}, \dots, w_0]_2$. In this method, the windows should not have the same size.

In [9] two types of windows are considered: zero windows formed only by bit 0, and the odd window of at most w length, beginning and ending with bit 1.

When reading binary image of number n “left to right” [9]

$$x^n = \left(\left(\left(\dots \left(\left(x^{(w_{i-1})_2} \right)^{2^{|w_{i-2}|}} \cdot x^{(w_{i-2})_2} \right)^{2^{|w_{i-3}|}} \dots \right)^{2^{|w_1|}} \cdot x^{(w_1)_2} \right)^{2^{|w_0|}} \cdot x^{(w_0)_2} \right) \quad (6)$$

When reading “right to left” [9]

$$x^n = \prod_{i=0}^{l-1} x^{(w_i)_2 \cdot 2^i} = \prod_{w \in \{1,3,\dots,2^w-1\}} \left(\prod_{\{i|(w_i)_2=w\}} x^{2^i} \right)^w, \quad (7)$$

where $l_i = \sum_{j=0}^{i-1} |w_j|$, for any $1 \leq i \leq l - 1$, $l_0 = 0$.

The Laboratory of RSA Data Security offers the following measurements to improve the resistance to such type of attack [12]:

1. ensuring continuous execution time of all modular exponentiation, however in this case the productivity of defence system reduces;
2. entering additional delays to the algorithm, however if there are not enough added delays a violator still can perform the cryptanalysis using additional measurements;
3. masking, that is multiplying the ciphertext by a random number to the implementation of exponentiation. This method of counteracting to timing attack reduces the performance of information security system by 2–10 %.

Thus there is a need to develop a new approach of asymmetric cryptographic protection from timing analysis implementation without sacrificing performance.

3 Fuzzy Selection System of Modular Exponentiation Method

Latest researches have shown that the modern attacks on implementation, especially passive attack of time analysis, are the most dangerous type of attacker’s illegal actions. Therefore, modern computer protection systems should provide robust resistance to time analysis, without reducing performance and memory cost.

As noted above, the main operation that affects the stability and performance of asymmetric cryptosystem is modular exponentiation. Choosing the method exponentiation by modulo, which resists time analysis and provides high system performance, is the priority task.

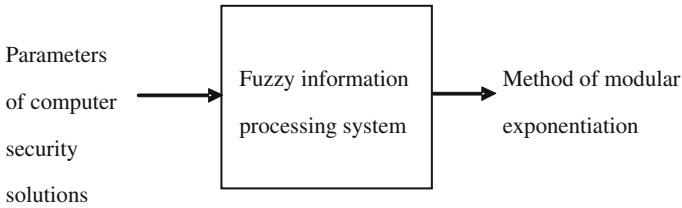


Fig. 2 General scheme of optimal selection of modular exponentiation method for distribution of computer system users' access

Considering the principle of Kochera [5], according to which the violator knows all about the encryption algorithm, except the key, as well as the actual implementation process of timing analysis that does not require the solving of the factorization problem, modular exponentiation exercise should be performed by means of individual method for each client.

In general the scheme of modular exponentiation choice method is shown in Fig. 2. In this case, the selection criteria are the basic parameters of the computer system of information protection, and the subsystem of the selection is the system of fuzzy information processing. The result of this system is one of the modular exponentiation methods that corresponds the input criteria of selection, using which computer system will ensure its optimum performance.

Since the distribution of users' access should consider the current system parameters such as performance, allowable memory consumption and the required level of timing analysis tolerance as well as fuzzy information about users, in order to solve this problem it is necessary to apply fuzzy logic [11].

For engineering problems Mamdani fuzzy mechanism is usually applied [11]. It uses a mini-max formulation of fuzzy sets. This mechanism includes the following steps [11]:

1. fuzzification procedure: degrees of truth are determined, that is the value of a membership function $MF_i(x)$ for the left part of each i rule (prerequisites);
2. fuzzy inference. Originally minimum "cut-off" level is defined for the left part of each of the rules $A_i = \min(MF_i(x))$, and then "truncated" membership functions of the conclusion $B_i = \min(A_i, B_i)$ are defined;
3. the composition or combining of received "truncated" function, where the maximum composition of fuzzy sets $MF(y) = \max(B_i(y))$ is used;
4. defuzzification or bringing to clarity. There are several methods of defuzzification, for instance, the method of the middle point or centroid method. The geometrical meaning of this value is the gravity center for curve function according to the obtained exit.

Application of fuzzy logic in the creation of hardware and software means to implement the access distribution in the computer system will ensure cryptosystem resistance to time analysis in real time. Given access distribution is carried out by

selecting the optimal modular exponentiation method for each client and incorporating current parameters of CS.

The main criteria of a computer system working capacity are a high performance, optimal memory consumption and resistance to malicious attacks.

In order to perform the transmission of information computer system uses the network for user accessing. Such a data network can be divided into protected and unprotected parts (Fig. 3).

In an unprotected network users K_1, K_2, \dots, K_n can be random, for this reason they are not reliable in a server from a security point of view, thus there is a high probability of a user being a violator. In addition, this part of the network is usually not protected against failures due to environmental effects and is open for all types of contemporary attacks on implementation.

In a protected part of the network (see Fig. 3) users $K_{i1}, K_{i2}, \dots, K_{im}$ are believed to be reliable and with regard to the security policy the existence of an internal violator is excluded. However, in this part of the network the possibility of passive timing analysis attack still remains [8, 13].

The server of a computer system consists of user identification subsystem, command subsystem and information processing unit (Fig. 4).

A user identification subsystem provides data to a processing unit about the required level of resistance to the timing analysis, including all the data about the user.

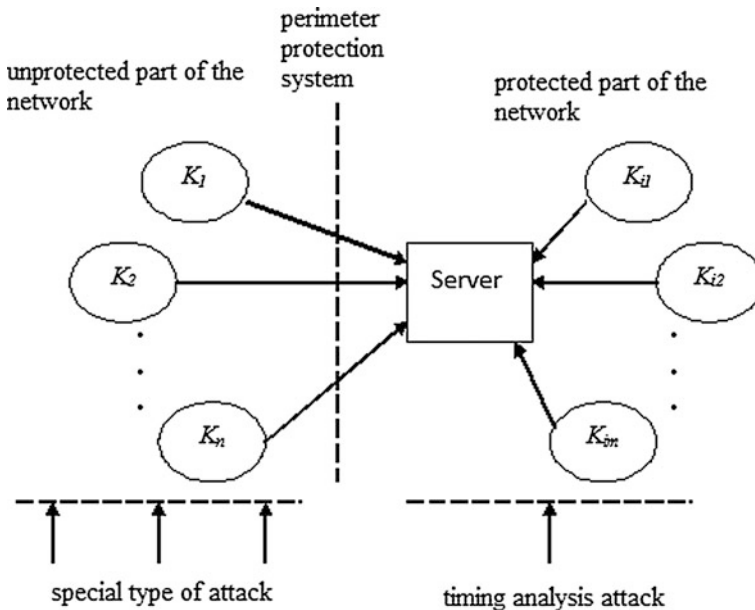


Fig. 3 The scheme of possible attacks on data transfer implementation in a computer system

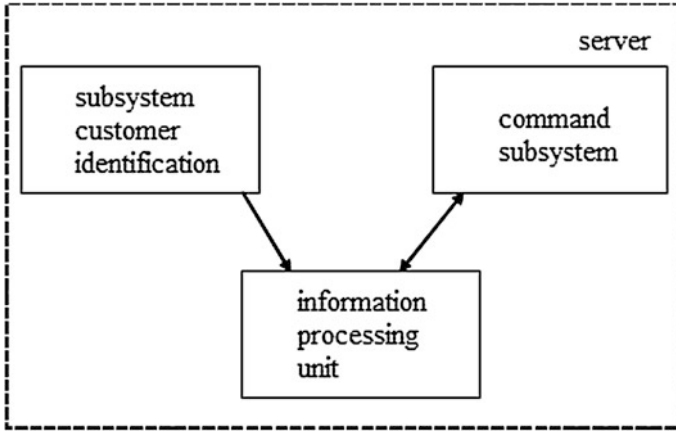


Fig. 4 The structure of a server of computer system

Users are known to a server by its IP-address and considering the “experience” of using the network they have their own level of reliability that can be set as the probability of failures in the transmission of information packets.

So, if a user is new to the system or has a very low level of reliability, the required level of resistance to timing analysis should be maximized. And vice versa for a user with a very high level of reliability the value of sustainability can go to 0, what will improve system performance.

Command subsystem of a server (see Fig. 4) provides a processing block with information about the computer system, namely the allowable memory consumption and the required level of performance.

In order to protect information in the network it is necessary to choose an optimal method of exponentiation by module to encrypt information or perform user authentication using cryptographic algorithm RSA. This problem is solved by an information processing unit, which is based on fuzzy logic, namely the mechanism of Mamdani fuzzy conclusion [11]. It handles input values of performance, memory consumption and resistance to timing analysis and provides an optimal method of modular exponentiation in each case to command subsystem which uses it to encrypt information. The main advantage of this unit is that it works in real time ensuring higher resistance of the system to malicious attacks as a violator will not know exactly the encryption algorithm [9, 12].

The general scheme of access distribution in a computer system is presented in Fig. 5.

A data processing block based on fuzzy logic is the basis of computer system protection. It is provided with selection criteria of modular exponentiation method, including the necessary level of resistance to the timing analysis R , productivity of cryptosystem P and the allowable memory consumption of the server M . Incoming fuzzy data are processed by a subsystem of optimal selection modular exponentiation method based on the mechanism of fuzzy conclusion according to Mamdani.

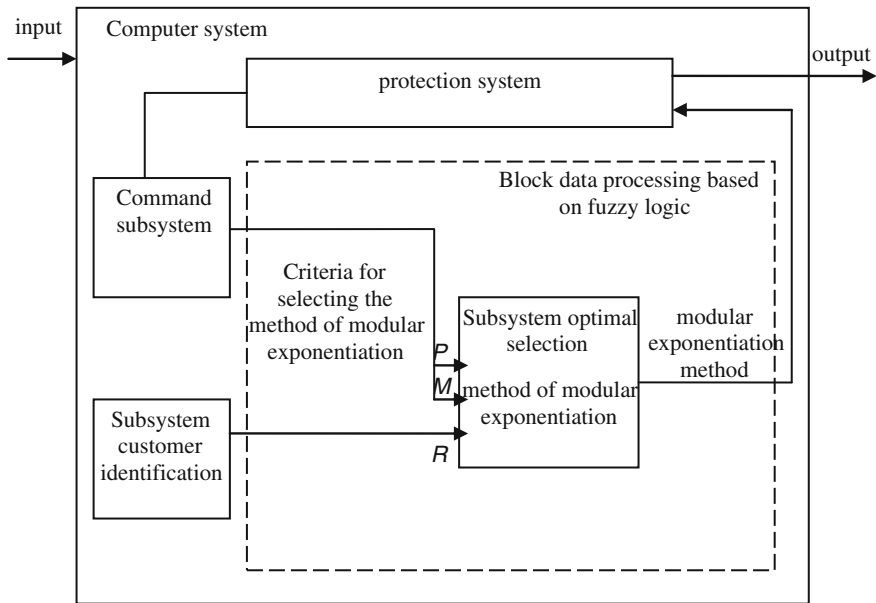


Fig. 5 General scheme of access distribution in a computer system based on fuzzy logic: P performance, M allowable memory expenses, R normalized resistance to timing analysis

The output of information by a processing block is a method of modular exponentiation that provides an optimal configuration of protection system regarding input selection criteria.

4 Conclusions

Thus, the structure of a computer network server introduced in the article helps to provide its optimal performance by means of access distribution to information resources. Further research of described fuzzy systems of the access distribution and its implementation on modern FPGA will allow us to realize a robust computer system against unauthorized access.

References

1. Bellezza, A.: Countermeasures against side-channel attacks for elliptic curve cryptosystems. In: Cryptology ePrint Archive, 2001/103 (2001). <http://citeseer.ist.psu.edu/bellezza01countermeasures.html>. Accessed 14 Sept 2014
2. Biham, E., Shamir, A.: Differential fault analysis of secret key cryptosystems. In: Kaliski B.S. Jr. (ed.) CRYPTO'97, 17th Annual International Cryptology Conference on Advances in

- Cryptology, Santa Barbara, CA, August 1997, pp. 513–525. Lecture Notes in Computer Science, No 1294. Springer, Heidelberg, (1997)
3. Convention on Cybercrime CETS No 185 The Cybercrime Convention Committee (T-CY), Strasbourg (2001). <http://conventions.coe.int/Treaty/en/Treaties/html/185.htm>. Accessed 14 Sept 2014
 4. Comer, D.E., Stevens, D.L.: Internetworking with TCP/IP, Vol III: Client-Server Programming and Applications. Prentice Hall, Upper Saddle River, NJ (2000)
 5. Gorbenko, I.D., Gorbenko, Y.I.: Prykladna kryptologia (Applied Cryptology). Fort, Kharkiv (2012)
 6. Kshetri, N., Murugesan, S.: EU and US Cybersecurity strategies and their impact on businesses and consumers. *Computer* **10**(46), 84–88 (2013)
 7. Kurose, J.F., Ross, K.W.: Computer Networking: A Top-Down Approach, 6th edn. Addison-Wesley, Boston (2011)
 8. Mangard, S., Oswald, E., Popp, T.: Power Analysis Attacks: Revealing the Secrets of Smart Cards. Springer, Berlin (2007)
 9. Moldovyan, A.A., Moldovyan, N.A., Sovyetov, B.A.: Kryptografiya (Cryptography). Lan', St Petersburg (2000)
 10. Muir, J.: Techniques of Side Channel Cryptanalysis. Technical Report, Department of Combinatorics and Optimization, University of Waterloo (2001)
 11. Shtovba, S.D.: Vvedeniye v teoriyu nyechotkikh mnozhestv i nyechotkuyu logiku (Introduction to the Theory of Fuzzy Sets and Fuzzy Logic). In: MATLAB. Exponenta (2001). <http://matlab.exponenta.ru/fuzzylogic/book1/>. Accessed 14 Sept 2014
 12. Stallings, W.: Cryptography and Network Security: Principles and Practice, 6th edn. Prentice Hall, Upper Saddle River, NJ (2013)
 13. Vasyiltsov, I.V.: Ataky specialnogo vydu na kryptoprystroyi ta metody borot'by z nymy (The Attacks of Special Type on Crypto Devices and Methods of Dealing with Them). KOHPI, Kremenets (2009)

Problem of Medicines Distribution on the Example of Pharmaceutical Wholesale

M. Cieśla and B. Mrówczyńska

Abstract The paper presents the task of determining the optimal route for the transport of medicines to pharmacies (i.e. traveling salesman problem) with two methods. The first one is a branch and bound method, the second is an artificial immune system. The calculations were performed with a TSPSG computer application for the first method and their own program for the other. In this way it was possible to conduct a comparative analysis of the ideal state, the optimum from the point of view of the results obtained, with the state of the current routing of medicines to pharmacies. This allowed us to verify the existing distribution channels in the logistics distribution of medicines.

Keywords Traveling salesman problem (TSP) · Vehicle routing problem (VRP) · Medicines transportation · Routing of medicines · Branch and bound method · Artificial immune system

1 Introduction

The idea of distribution logistics is to ensure the availability of goods in places where there is a demand, while also putting emphasis on cost optimization of the process and the appropriate level of customer service. Guideline for all logistics activities is to provide a product to the right customer at the right time and place, in the required quality and quantity, at the right cost [1].

Many commercial and industrial enterprises face the challenge of a proper organization of the distribution of its products to a network of customers. In the field of operations research, this problem is known as the vehicle routing problem (VRP). A characteristic feature of transport routing problems is that using it enables

M. Cieśla (✉) · B. Mrówczyńska

Faculty of Transport, Silesian University of Technology, Gliwice, Poland
e-mail: Maria.Ciesla@polsl.pl

B. Mrówczyńska

e-mail: Bogna.Mrowczynska@polsl.pl

one to formulate any problem, as opposed to actually finding its best solution. Therefore, these issues are of interest to many researchers, resulting in a large number of proposals for algorithms to find the best solutions.

2 Specificity of Distribution in the Pharmaceutical Industry

The literature reveals a number of distribution definitions. Basically, they can be classified into two main perspectives: marketing and logistics.

According to A. Czubała, distribution of products is referred to as “profit-oriented activities including planning, organization and control of how to deploy products and offer them for sale.”

According to H.Ch. Pfohl: “Distribution logistics is a part of logistics system, associated with the market. It combines the company’s production logistics with procurement logistics of the buyer. Distribution logistics covers all activities that are related to supplying a customer with finished goods.”

The purpose of the distribution is to provide purchasers of the final product desired by them to places where they want to buy them in the corresponding time on the agreed terms and the price accepted by them. This task requires the manufacturer to decide on a choice of distribution channels, as well as how to choose the physical movement of products. Distribution channels of appropriate capacity, structure and competitiveness should enable the manufacturer to estimate a planned sales volume, while the corresponding physical distribution organization must provide an adequate level of service to buyers while minimizing distribution costs. Figure 1 shows the structure of distribution [2].

Supply medicines (manufactured and destined for sale) never coincide with demand (real demand). The main differences relate to, among other considerations, place, time of production and consumption, type and range of medicines, as well as the size of single batches of different supplies. Elimination of these differences is leading to increasingly important decisions regarding the development of distribution channels and physical processes of medications marketing [3].

Presenting characteristics of different distribution channels in the pharmaceutical market, diversity of intermediaries in the horizontal and vertical channels should be emphasized. The system shows all possible forms and types of mediation: pre-wholesale, wholesale, small brokers, retailers and patients as terminal buyers (Fig. 2—on the basis of [1]).

In the pharmaceutical industry there is also the phenomenon of parallel imports of medicines. A scheme of this kind of trade is shown in Fig. 3 [4]. The main reason for the emergence of a form of commerce (a variety of territorial expansion) is the difference of prices of drugs that are on the markets of countries belonging to the European economic zone. The principle of free movement of medicines in force in the European Union allows the flow of medicines from countries with lower

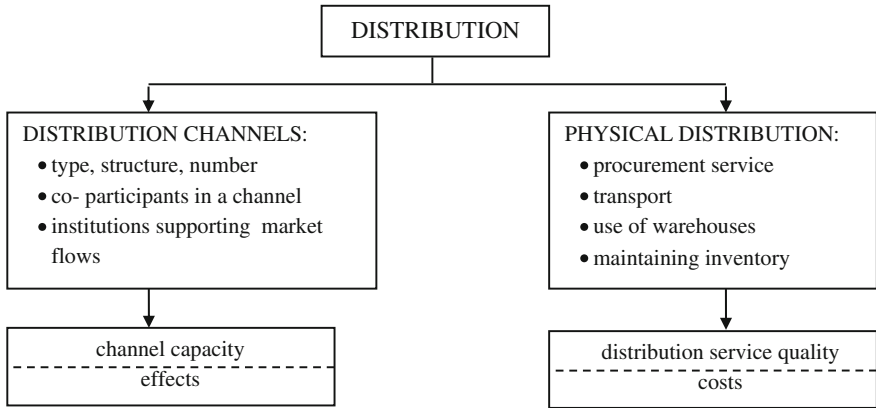


Fig. 1 Distribution processes structure

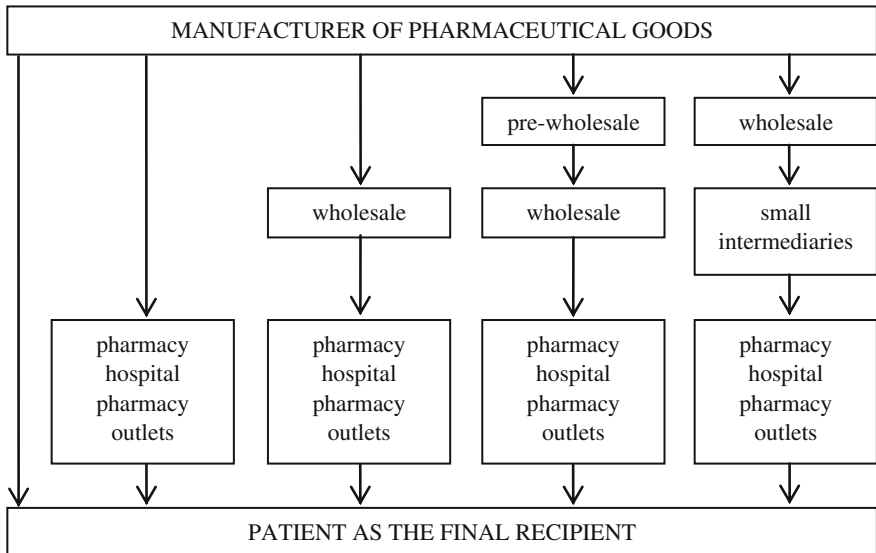


Fig. 2 Schematic distribution channels in the pharmaceutical industry

prices to countries where prices are at higher levels. Parallel trade in medicines on the circulation of goods protected by patents in a given market by third parties that distribute the goods to the territory of another country without the authorization of the rights of the manufacturer, often against his will, and regardless of the distribution channel. It is estimated that currently medicines derived from parallel trade constitute about 6 % of the total pharmaceutical market in the European Union [4].

This article focuses on activities related to bringing medicines to pharmacies. They are based on the so-called physical processes in distribution of the flow, and

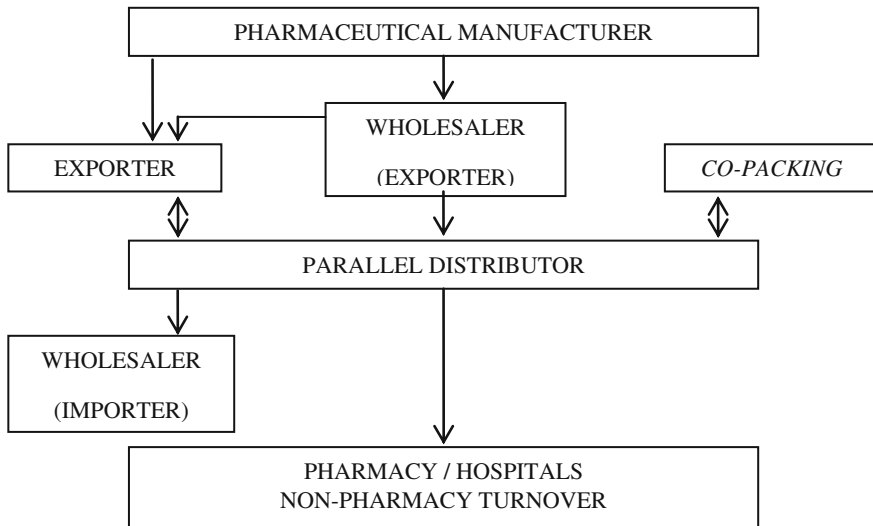


Fig. 3 Channel of medicines parallel import in the pharmaceutical market

hence the selection of respiratory movement of goods and the type of cells that mediate the flow of goods from producer to consumer.

3 Example of Route Optimization of Medicines Physical Distribution

In the field of operations research, the problem of route optimization is called the Vehicle Routing Problem. A characteristic feature of the vehicle routing problem is the ease of problem formulation, as opposed to its solution. Therefore, these issues are of interest to many researchers, resulting in a large number of proposals for algorithms searching for the best solutions.

An exemplary case study was carried out on routes for the distribution of medicines in Bielsko-Biala. Along the route a driver has 29 points of receipt of goods (pharmacy). The place of loading is a pharmaceutical wholesaler “Medicare” located in Myslowice-Kosztowy. Wholesale is the thirtieth point in the analysis [5]. The current route is determined on the basis of the following limitations:

- pharmacy opening hours,
- designated prohibition time zones of entry (pedestrian, market),
- distance between pharmacies.

Table 1 shows the order according to which the driver visits customers. It also includes distance and travel times between itinerary points, as well as the total length and travel time.

Table 1 Current system medicines distribution on the route Myslowice—Bielsko-Biala—Myslowice

No.	Pharmacy	Distance from the previous (km)	The average travel time (min)
1	“Ziko” Ul. Warszawska 5	54.0	51
2	“Klimczok” Ul. Cyniarska 11	1.5	3
3	“W Białej” Ul. 11 Listopada 21	0.3	1
4	“Ziko” Ul. 11 Listopada 37	0.3	1
5	“Św. Anny” Ul. Stojałowskiego 55	0.6	2
6	“Bielska” Plac Żwirki i Wigury 9	1.2	3
7	“Ziko” Ul. Leszczyńska 20	3.5	8
8	“Beskid” Ul. Sternicza 26a	3.0	7
9	“Słoneczna” Ul. Sobieskiego 64a	2.7	6
10	“Pod Zamkiem” Ul. Mickiewicza 6	1.6	4
11	“Pod Akacjami” Ul. Teodora Sixta 11	0.7	2
12	“Ziko” Ul. Mostowa 5	1.1	2
13	“Ogólnodostępna” Ul. Barlickiego 2	0.5	2
14	“Pod Magurką” Ul. Żywiecka 92	2.5	5
15	“Złote Łany” Ul. Jutrzenki 24	1.0	3
16	“Złote Kłosa” Ul. Siewna 2f	1.0	4
17	“Agaty” Ul. Górska 158	3.4	8
18	“Na Błoniach” Ul. Willowa 2a	5.3	10
19	“Pod Dębowcem” Al. Armii Krajowej 132	3.6	9
20	“Elena” Ul. Szarotki 10	4.3	8
21	“Konfarm” Ul. Pszenna 11	1.2	3
22	“Centauria” Ul. Spółdzielców 50	0.9	2
23	“W Wapienicy” Ul. Cieszyńska 421	2.5	5
24	“WCL” Ul. Cieszyńska 418	0.1	1
25	“Małgorzata” Ul. Międzyrzecka 7	0.3	1
26	“Przy Wiarusie” Ul. Bielska 38, Jaworze	3.1	5
27	“Moja” Ul. Bohaterów Monte Cassino 421	8.0	6
28	“Galen” Ul. Warszawska 180	5.5	5
29	“Beauty Pharm” Ul. Mostowa 2	3.5	6
30	Medicare Sp. z o. o.	55.0	49
	Total	172.2	222

The current drive is carried out daily on the route Myslowice—Bielsko-Biala—Myslowice and is 172.2 km. The loading operations include invoice scanning and loading of goods. The scanning time of invoices is about 15 min and the cargo loading time is about 45 min. The average service time at each point of reception is 5 min. Unloading of empty containers after returning to the warehouse is about 15 min. The total work time is approximately 442 min.

3.1 *The Mathematical Model of the Traveling Salesman Problem*

The driver leaves the warehouse, which is a loading place, to visit a number of its customers (pharmacies). All pharmacies must be visited exactly once and then the truck is returned to the warehouse (place of loading). The order of visits to each customer is free, but the driver's aim is to make the route as short as possible. Such a task is a special case of VRP and is called the traveling salesman problem (TSP).

A convenient tool for description of the traveling salesman problem is to provide, in the form of a graph, a network of routes that the van may possibly pass through. Such a graph is a mixed graph $\Gamma = (V, E, A)$, where V is a non-empty set of vertices, and E is the set of edges—two-element subsets of vertices V , and A is the set of arcs, which are elements of the Cartesian product: $V \times V = \{(a, b): a \in V, b \in V\}$, where a is the initial vertex, and b is the end vertex. Graph vertices frequently correspond to the crossroads. Sections of two-way traffic between the intersections are the edges, and sections of one-way streets are the arches directed according to the direction of travel. Weighting of the edges and arcs in the example shown will be equal to the lengths of the respective sections of roads, and transit times. Figure 4 shows a map section of Bielsko-Biala with selected pharmacies and a graph representing a route network that is possible to pass by all relevant customers. A solution to the Travelling Salesman Problem is equivalent to finding a Hamiltonian cycle in the graph.

But to find such a cycle, you should make sure that the relevant graph is of Hamiltonian type. One of the fundamental statements about this issue is Ore's theorem [6] which states that if a graph G is an undirected graph with n vertices, where $n \geq 3$, and for each pair of non-ordered vertices a and b , the sum $\rho(a) + \rho(b) \geq n$, where ρ is the degree of the vertex [7]. The graph considered in this article has 30 vertices. But as it can be easily seen in Fig. 4, no vertices have a degree higher than 6. So the condition of the Ore theorem cannot be met. A possible solution is to extend the route network graph to a complete graph. The degree of each vertex of the complete graph is equal to $n - 1$. A complete graph is Hamiltonian type and is designed to determine the $(n - 1)!$ Hamilton cycles. This is a NP complete problem. Bringing graph Γ to the G complete graph is performed as follows: for any two vertices in the graph, the shortest distances are determined using Dijkstra's algorithm. Any two vertices were combined with an edge or an arc length corresponding to the shortest path from one vertex to another. The results are summarized in Table 2.

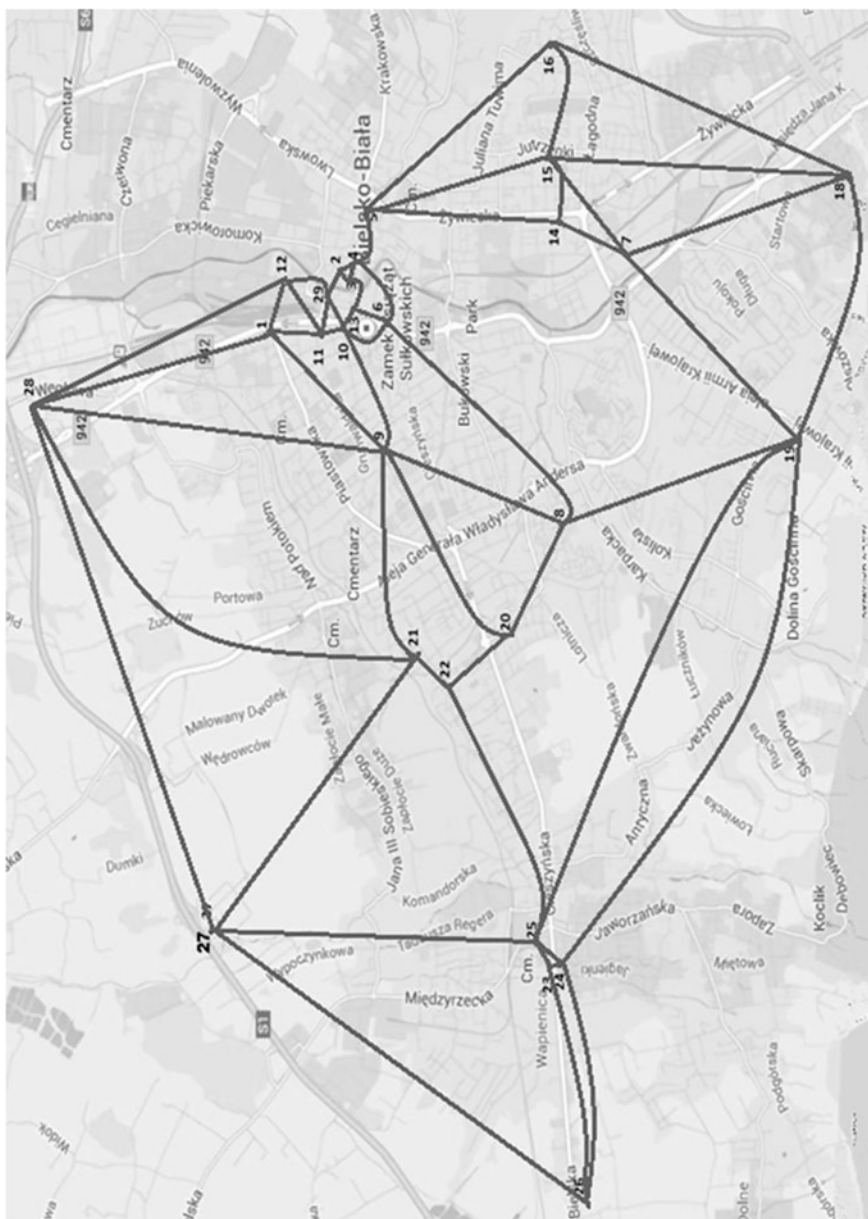


Fig. 4 Channel of medicines parallel import in the pharmaceutical market

Table 2 Distance matrix between wholesale customers

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0.0	54.0	58.0	57.0	58.0	58.0	55.0	57.0	59.0	56.0	55.0	55.0	55.0	56.0	57.0
2	54.0	0.0	1.3	1.6	1.6	1.7	1.3	3.2	4.0	1.9	1.0	0.9	1.1	1.0	3.4
3	58.0	1.3	0.0	0.3	0.3	1.5	1.4	2.8	4.1	2.0	1.1	1.8	0.1	1.0	2.3
4	57.0	1.6	0.3	0.0	0.3	0.9	1.1	2.8	4.0	1.7	1.3	1.5	0.6	0.6	2.2
5	58.0	1.6	0.5	0.3	0.0	0.6	1.2	2.5	3.9	1.6	1.4	1.5	0.6	0.6	2.0
6	58.0	1.7	1.5	0.9	0.6	0.0	1.2	2.6	4.2	1.8	1.1	1.8	0.8	0.7	2.2
7	55.0	1.3	1.4	1.1	1.2	1.2	0.0	3.5	3.0	1.9	1.0	1.1	1.1	0.4	2.2
8	57.0	3.2	2.8	2.8	2.5	2.6	3.5	0.0	3.0	3.0	3.0	3.1	3.7	3.3	0.8
9	59.0	4.0	4.3	4.0	3.9	4.2	3.0	3.0	0.0	2.7	4.0	4.0	4.7	4.5	2.9
10	56.0	1.9	2.0	1.7	1.6	1.8	1.9	3.0	2.7	0.0	1.6	1.4	2.4	1.2	3.0
11	55.0	1.0	1.7	1.3	1.4	1.7	1.0	3.0	4.0	1.6	0.0	0.7	1.6	1.6	3.5
12	54.0	0.9	1.8	1.5	1.5	1.8	1.1	3.1	4.0	1.4	0.7	0.0	1.1	0.9	3.6
13	55.0	1.1	0.3	0.6	0.6	0.8	1.7	3.7	4.7	2.4	1.6	1.1	0.0	0.5	2.9
14	55.0	1.0	1.0	0.6	0.6	0.7	0.4	3.3	4.5	1.2	1.6	0.9	0.5	0.0	2.5
15	57.0	3.4	2.3	2.2	2.0	2.2	2.2	0.8	2.9	3.0	3.5	3.6	2.9	2.5	0.0
16	58.0	3.9	2.8	2.7	2.6	2.2	2.6	1.3	3.4	3.5	4.0	4.1	3.4	3.0	1.0
17	58.0	4.4	3.3	3.2	3.1	3.2	4.1	2.3	4.0	4.1	4.6	4.7	4.0	3.5	2.0
18	67.0	7.6	6.0	6.0	5.9	5.7	6.5	3.6	53.0	5.4	5.9	6.0	5.3	6.8	3.8
19	64.0	6.0	6.0	5.7	5.6	5.7	5.8	2.7	5.0	5.6	6.1	6.2	6.4	6.2	3.6
20	59.0	5.1	5.2	4.8	4.7	4.8	5.7	2.5	4.0	5.0	5.6	5.7	5.9	5.7	2.8
21	57.0	3.5	4.4	4.0	3.9	4.0	3.8	4.5	1.5	2.0	3.4	3.5	4.5	4.4	4.9
22	58.0	4.0	53.0	5.0	4.9	5.0	4.8	5.0	2.0	2.5	3.9	4.0	4.5	4.9	5.4
23	58.0	4.0	5.2	4.9	4.8	4.9	4.7	4.9	1.9	2.5	3.9	4.0	5.0	4.8	5.3
24	60.0	5.1	6.9	6.6	6.5	6.1	6.5	6.7	3.7	4.2	5.6	5.1	1.3	7.0	7.1

(continued)

Table 2 (continued)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
25	60.0	5.7	6.9	6.6	6.5	6.7	6.5	6.7	3.7	5.6	5.7	7.3	7.0	7.1
26	60.0	5.6	6.9	6.5	6.4	6.6	6.4	6.6	3.6	5.6	5.1	7.2	6.9	7.0
27	64.0	7.9	9.2	8.9	8.8	9.0	8.8	8.9	5.9	7.8	1.9	9.4	9.1	9.3
28	56.0	7.9	11.5	11.2	11.1	11.3	11.1	11.3	8.3	10.1	10.6	10.8	10.5	10.3
29	53.0	3.0	3.1	3.4	3.3	3.5	3.1	5.0	6.1	2.9	2.8	3.0	3.2	5.2
30	58.0	0.7	0.6	0.5	0.6	0.9	0.9	3.0	3.4	1.4	0.8	0.1	0.5	2.8
	16	17	18	19	20	21	22	23	24	25	27	28	29	30
1	58.0	58.0	61.0	64.0	59.0	57.0	58.0	58.0	60.0	60.0	64.0	58.0	53.0	58.0
2	3.9	4.4	7.6	6.0	5.1	3.5	4.0	4.0	5.7	5.6	7.9	7.9	3.0	0.7
3	2.8	3.3	6.0	6.0	5.2	4.4	5.3	5.2	6.9	6.9	9.2	11.5	3.7	0.6
4	2.7	3.2	6.0	5.7	4.8	4.0	5.0	4.9	6.6	6.5	8.9	11.2	3.4	0.5
5	2.6	3.1	5.9	5.6	4.7	3.9	4.9	4.8	6.5	6.4	8.8	11.1	3.3	0.5
6	2.2	3.2	5.7	5.7	4.8	4.0	5.0	4.9	6.7	6.6	9.0	11.3	3.5	0.6
7	2.6	4.1	6.5	5.8	5.7	3.8	4.8	4.7	6.5	6.4	8.8	11.1	3.1	0.9
8	1.3	2.3	3.6	2.7	2.5	4.5	5.0	4.9	6.7	6.6	8.9	11.3	5.0	3.0
9	3.4	4.0	5.3	5.0	4.0	1.5	2.0	1.9	3.7	3.6	5.9	8.3	6.1	3.4
10	3.5	4.1	5.4	5.6	5.0	2.0	2.5	2.5	4.2	4.1	6.4	5.4	3.6	2.3
11	4.0	4.6	5.9	6.1	5.6	3.4	3.9	3.9	5.6	5.6	7.8	10.7	2.9	1.4
12	4.1	4.7	6.0	6.2	5.7	3.5	4.0	4.0	5.7	5.7	7.9	10.6	2.8	0.8
13	3.4	4.0	5.3	6.4	5.9	4.5	45.0	5.0	7.1	7.2	9.4	10.8	3.0	0.1
14	0.3	3.5	6.8	6.2	5.7	4.4	4.9	4.8	1.0	6.9	9.1	10.5	3.2	0.5
15	1.0	2.0	3.8	3.6	2.8	4.9	5.4	5.3	7.1	7.0	9.3	10.3	5.2	2.8
16	0.0	1.0	3.3	4.2	3.3	5.4	5.9	5.8	7.6	7.5	9.8	10.8	5.7	33.0
17	1.0	0.0	3.4	5.1	4.3	6.0	6.4	6.4	8.2	8.1	10.4	11.4	6.1	3.9

(continued)

Table 2 (continued)

16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
18	3.3	3.4	0.0	5.3	6.4	8.2	8.8	10.4	10.4	10.3	12.3	15.6	9.9	6.4
19	4.2	5.1	5.3	0.0	3.6	6.7	73.0	8.9	8.9	8.8	11.1	10.8	7.8	45.0
20	3.3	4.3	6.4	3.6	0.0	4.3	5.5	5.5	5.5	5.4	7.7	8.0	6.3	3.1
21	5.4	6.0	8.2	6.7	4.3	0.0	1.2	2.6	2.6	2.5	4.8	6.2	5.4	4.6
22	5.9	6.4	8.8	7.3	5.5	1.2	0.0	3.4	3.4	3.3	5.7	7.9	5.9	5.4
23	5.8	6.4	8.8	7.3	5.1	0.9	0.9	2.5	2.4	2.3	4.7	6.0	5.9	55.0
24	7.6	8.2	10.0	8.9	5.5	2.6	3.4	0.0	0.1	0.3	2.2	4.0	9.8	7.1
25	7.6	8.2	10.0	8.9	5.5	2.6	3.4	2.5	0.1	0.0	2.2	4.0	9.8	7.1
26	7.6	8.1	10.0	8.8	5.4	2.5	3.3	0.3	0.3	0.0	3.1	3.7	9.5	7.0
27	9.8	10.4	12.3	11.1	7.1	4.8	5.1	2.2	2.3	3.1	0.0	8.0	13.8	16.5
28	10.8	11.4	15.6	10.8	8.0	6.2	7.9	4.0	2.6	3.7	8.0	0.0	5.5	11.0
29	5.7	6.1	9.9	7.8	6.3	5.4	5.9	9.8	9.8	9.5	13.8	5.5	0.0	3.5
30	3.3	3.9	6.4	4.5	3.1	4.6	5.4	7.1	7.1	7.0	16.5	11.0	3.5	0.0

The analysis presents the Traveling Salesman Problem for one of the 43 routes of suppliers from “Medicare” pharmaceutical wholesalers.

As already mentioned above, the TSP presented is of the combinatorial nature and belongs to the NP-complete problems. Despite a very simple formulation of the Traveling Salesman Problem, finding the optimal solution (the shortest route) with a large number of customers, all of which the driver has to visit, it is extremely time-consuming. Therefore, methods used are less accurate, but gives a good enough solution in a relatively short period of time.

This paper presents two methods to solve the problem: the branch and bound method and an artificial immune system.

3.2 Route Selection with Branch and Bound Method

First the problem was solved with the branch and bound method usually used for discrete and combinatorial optimization problems or general real-valued problems. The algorithm of this method consists of the set of solutions which is thought of as forming a rooted tree with the full set at the root. The algorithm explores branches of this tree, which represent subsets of the solution set.

For the calculation with branch and bound method, the TSPSG program was used [8]. On the input there are the number of cities (points) and the distances between them in the form of a matrix of distances between wholesale clients (Table 2). The analysis of the transport route included 29 clients (points) and place of loading which is the pharmaceutical wholesaler.

After performing the calculations, a new order of carriage made in the TSPSG program can be presented as shown in Table 3. The length of the route designated by the program is 163.60 km. There are also given the distances between points and travel time. At the end of the array is given the sum of the distance and the driver travel time.

Total travel time is about 44 min shorter and had less complete combustion of 1.28 l/100 km. Theoretical calculations show that the route designated by the program TSPSG is shorter and more economical than the present, but due to limitations in practice, this route is sub-optimal for the journey. A TSPSG program does not include such restrictions as pharmacy opening hours, resulting in longer runs. For example, the opening time difference between the second, and third point generated by the program is 75 min. The biggest problem for the established order of the transport route is the delivery delay of drugs to pharmacies opened in earlier hours. An example is the point 29 Pharmacy “Sw. Anny”, which in the current route is supplied as fifth.

Table 3 Route order determined with branch and bound method using TSPSG program

Pharmacy	Distance from previous (km)	Average travel time (min)
“Ziko” Ul. Warszawska 5	54.0	51
“Beauty Pharm” Ul. Mostowa 2	0.7	1
“Na Błoniach” Ul. Willowa 2a	4.5	8
“Pod Dębowcem” Al. Armii Krajowej 132	3.5	6
“Ziko” Ul. Leszczyńska 20	2.5	3
“Pod Magurką” Ul. Źywiecka 92	0.7	1
“Złote Łany” Ul. Jutrzenki 24	1.0	2
“Złote Kłosy” Ul. Siewna 2f	1.0	2
“Agaty” Ul. Górska 158	3.4	5
“Słoneczna” Ul. Sobieskiego 64a	5.9	9
“Konfarm” Ul. Pszenna 11	2.5	3
“Beskid” Ul. Sternicza 26a	2.0	3
“Elena” Ul. Szarotki 10	1.5	2
“Przy Wiarusie” Ul. Bielska 38, Jaworze	4.8	6
“W Wapienicy” Ul. Cieszyńska 421	2.2	3
“Małgorzata” Ul. Międzyrzecka 7	0.2	1
“Moja” Ul. Bohaterów Monte Cassino 421	3.7	4
“Wapienickie Centrum Leków” Ul. Cieszyńska 418	2.6	4
“Centauria” Ul. Spółdzielców 50	2.4	3
“Pod Akacjami” Ul. Teodora Sixta 11	3.9	5
“Pod Zamkiem” Ul. Mickiewicza 6	0.7	1
“Bielska” Plac Źwirki i Wigury 9	1.0	2
“Ogólnodostępna” Ul. Barlickiego 2	0.4	1
“Ziko” Ul. Mostowa 5	0.5	1
“Klimczok” Ul. Cyniarska 11	0.3	1
“W Białej” Ul. 11 Listopada 21	0.3	1
“Ziko” Ul. 11 Listopada 37	0.3	1
“Św. Anny” Ul. Stojałowskiego 55	0.6	1
“Galen” Ul. Warszawska 180	3.5	4
Medicare Sp. z o. o. Ul. Białobrzaska 45, Mysłowice	53.0	50
Total	163.6	185.0

3.3 Route Selection with Artificial Immune System

An artificial immune system is used for the calculation methods of artificial intelligence and imitates the human immune system [9]. The natural immune system responds to attacks by antigens—foreign objects invading the body. There are several barriers that protect the body: skin, which protects not only mechanically,

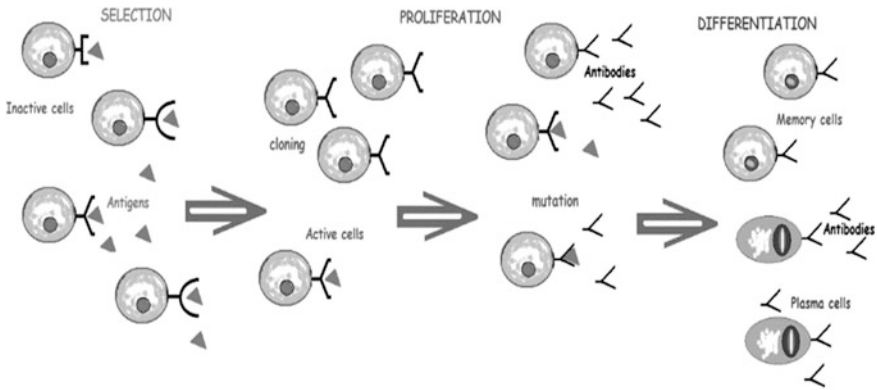


Fig. 5 Clonal selection

but for example chemically through sweat, or enzymes on the skin. The immune system is activated when the previously described defense mechanisms fail.

The body produces billions of antibodies. Many of them are mutated with the purpose of generating those that fit snugly against the antigen. This allows destruction of the antigen and removal of it from the body. After finding such an antibody, it takes a rapid proliferation of cells that produce best fitted antibodies. These cells are released into the bloodstream. After removing antigens follows suppression. The number of antibodies directed against a specific antigen is reduced considerably. In the next attack on the same antigens, the immune system will work faster. The cycle seeking a better and better antibody is called a clonal selection (Fig. 5).

The paradigm of clonal selection is used to optimize. The clonal selection algorithm proceeds as follows:

1. Initialization
Strings of numbers (2) are randomly generated, for which a fitting value is determined.
2. Cloning
The solutions of the highest fitting values are repeatedly cloned.
3. Mutation
Most of the clones obtained are subjected to mutation operators to change the order of the numbers in the sequence.
4. Suppression
The number of antibodies is limited to the number of primary population. A variety of solutions are maintained.

In the model of an artificial immune system, the antibody is a string of numbers representing the vertices, that is, in this example, the numbers of pharmacies listed in order of their supply:

$$[k_1, k_2, \dots, k_n] \tag{1}$$

where:

- n number of delivery points,
- k_i the next delivery points, $i = 1, \dots, n$

Mutating is to change the order of the vertices. For the objective function in the distance traveled optimization, and for an affinity function the inverse of the objective function is assumed. In this way, the function matches the higher value of the distance length for the specified route that is less [10].

For the calculations the following objective function was adopted:

$$f = w_1 \frac{\sum_{i=1}^n d_{k_i k_{i+1}}}{d_{max}} + w_2 \frac{\sum_{i=1}^n t_{k_i k_{i+1}} + t_{dt}}{t_{max}} \tag{2}$$

where:

- n number of delivery points,
- k_i, k_j the next delivery points; $k_i, k_j \in [k_1, k_2, \dots, k_n]$
- d_{k_i, k_j} distance from point k_i to k_j ,
- t_{k_i, k_j} travel time from point k_i to k_j ,
- w_1, w_2 weight coefficients,
- d_{max} maximum path,
- t_{max} maximum travel time,
- t_{dt} total delay time.

The total delay time is the sum of all delays. For example time windows in which another pharmacy is supposed to be supplied, are taken into account in such a way that if delay occurs the penalty for that is added to, and in case of supplier’s earlier arrival to the point, it is assumed that the driver is waiting for the time of scheduled opening time of pharmacy. Service time is calculated only from the opening time of each pharmacy. The restrictions were taken into account in the form of a penalty function, which is the size beyond the limits which are multiplied by coefficients of large numerical values.

The results of calculations are presented in Table 4. The shortest received route is a fourth route with a length of 157.2 km. The calculations do not take into

Table 4 Transport routes determined with artificial immune system

No.	Matching function	Path (km)	Route order
1	0.634115	157.7	30-28-1-10-6-13-15-17-16-8-20-21-22-25-23-24-26-11-9-5-29-12-2-3-4-14-7-18-19-27-30
2	0.618429	161.7	30-1-10-11-29-3-2-4-5-13-6-8-20-21-22-25-26-19-18-7-14-15-16-17-9-27-24-23-12-28-30
3	0.622278	160.7	30-1-10-6-13-5-4-3-2-12-29-11-9-7-19-18-17-16-15-14-8-21-20-25-27-24-23-26-22-28-30
4	0.636132	157.2	30-1-29-5-4-3-2-12-13-23-25-27-24-26-11-10-7-19-18-17-16-15-14-9-21-22-20-8-6-28-30

Table 5 Comparative analysis of routes determined with artificial immune system

Routes	Distance (km)	Travel time (min)	Fuel consumption (l) ^a
Current	172.2	222	14.64
Branch and bound method	163.6	185	13.91
Artificial immune system method	157.2	178	13.36

^aAssuming combustion of 8.5 l/100 km

account the restrictions on time windows for individual pharmacies but the algorithm, which was used to calculate, copes well with such problems.

4 Conclusions

Table 5 presents the final comparison of the current state with the propositions to distribute medicines in the distribution channel of case study, determined with the branch and bound method (using TSPSG application) and using an artificial immune system (own program).

The values shown in previous table indicate the following conclusions:

- best results were obtained using an artificial immune system,
- total distance of the best result is reduced by 15 km as compared to the status quo and 6.4 km resulting from the branch and bound method.

References

1. Bendkowski, J., Kramarz, M.: *Podstawy logistyki dystrybucji*. Wydawnictwo Politechniki Śląskiej, Gliwice (2003)
2. Czubała, A.: *Dystrybucja produktów*. PWE, Warszawa (1996)
3. Stern, L.W., Ansaray, A.J., Coughlan, A.T.: *Kanały marketingowe*. Wydawnictwo naukowe PWN, Warszawa (2002)
4. Michalik, M., Pilarczyk, B., Mruk, H.: *Marketing strategiczny na rynku farmaceutycznym*, Oficyna Wolters Kluwer Business, Wyd. II, Kraków (2008)
5. Włosek, J.: *Logistyka dystrybucji leków na przykładzie firmy "Medicare" Sp. z o. o.*, Bachelor of science (engineering) thesis written under the supervision of M. Cieśla; Wyższa Szkoła Biznesu w Dąbrowie Górniczej, Dąbrowa Górnicza (2014)
6. Wilson, R.J.: *Introduction to Graph Theory*. Longman Group Ltd. (1985)
7. Deo, N.: *Graph Theory with Application to Engineering and Computer Science*. Prentice-Hall, Englewood Cliffs (1974)
8. TSPSG application available on www.tspsg.info. Accessed on 12.10.2014
9. Castro, L.N., Zuben, F.J.: *Artificial immune systems, part I—basic theory and applications*. Technical Report, TR—DCA 01/99, December (1999)
10. Mrówczyńska, B.: *Optimal routes scheduling for municipal waste disposal garbage trucks using evolutionary algorithm and artificial immune system*. *Transp. Probl.* **6**(4), 5–12 (2011)

Part II
Modelling of Mechanical Systems

Bond Graph Model of a Robot Leg

B. Gola, J. Kopec, J. Rysiński and S. Zawislak

Abstract In the paper, we present the bond-graph model of a 12-legged robot. The robot was built within the framework of a master thesis. The system of control was designed and performed by the authors. There are versatile control options e.g. via a PC-type computer as well as commercial mobile devices such as tablets or mobile phones. Various tests of the robot were performed. The bond-graph model allows for versatile simulations that help a designer in finding the most efficient solution. The bond graph schemes are discussed in the paper.

Keywords Robot leg · Leg linkages · Drive · Bond graph models

1 Introduction

Legged robots have various applications and they are widely researched nowadays. In the present paper we consider the 12-legged robot [1]. It was built and tested in different walking routines.

A leg is an important robot part. The leg design options and operation possibilities have crucial importance for robot mobility and the velocity of robot walking.

In the present paper, the bond graph model of the robot leg is considered. Several methods of modeling of robots' activities are utilized. Graph-based approaches have been used in recent years due to its generality and usefulness for

B. Gola · J. Kopec · J. Rysiński (✉) · S. Zawislak
University of Bielsko-Biala, Bielsko-Biala, Poland
e-mail: jrysinski@ath.bielsko.pl

B. Gola
e-mail: bartlomiejgola@wp.pl

J. Kopec
e-mail: jkopec@ath.bielsko.pl

S. Zawislak
e-mail: szawislak@ath.bielsko.pl

the analyses of versatile mechanisms. Different problems related to mechanisms and robots were solved by means of different types of graphs e.g. bond graphs [2, 3], linear graphs and directed graphs [4–6] as well as modular graphs [7], respectively. The general information on bond graph modeling and linear graph modeling of chosen mechanical systems can be found in books [8, 9, 10], respectively. The problems relevant to the robotics field that were solved by means of graphs are e.g.: navigation [11], predictive control [12], robot configurations analysis [13, 14], optimization [13–15] etc. Usefulness of graph modeling for gears and other mechanical systems was confirmed in the following papers [4, 16, 17]. Bond graphs are used frequently for robot modeling [1, 2, 5, 18–21]. In other cases of modeling the general rules of mechanics are utilized [22] or specialized approaches such as e.g. screw theory [23]. Frequently robot limbs, especially legs, are analyzed [2, 15, 20, 24].

The DUODEPED robot is described in the present paper. The robot was built within the framework of a diploma thesis prepared by Gola [1]. The leg design is based upon an idea of Theo Jansen, a Dutch scientist who presented a similar solution in 2005 in the form of a “beach beast”, an artificial creature driven by wind. However, in the case we are discussing, the design and all of the technical drawings were done by diploma students and the robot was driven by an electric motor.

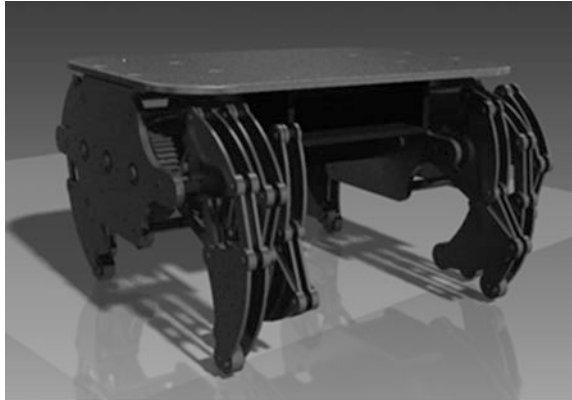
2 General Description of the Robot

As stated above, the robot was designed and manufactured at the University of Bielsko-Biala to fulfill the needs for a master of mechanical engineering. The DUODEPED is a robot of two independent degrees of freedom. The mechanical couplings between the robot legs on each side enable movement onward (ahead), backwards as well as turning and rotating. The control system of the considered robot is similar to the control system of caterpillar robots with independent controlled tracks. A robot of this design can only walk in one manner, however the velocity can be controlled effectively by the control elements.

There were three stages of robot development:

- (i) design of chassis, legs and leg fixing points, driving system and accomplishing of technical drawings of all elements using 3D Software, Inventor (Fig. 1);
- (ii) design of an electronic system, main board, Bluetooth communication unit, USB/RS-232 interface unit, CD motor controller. All robot electric circuits were imprinted (in the laboratory of the University of Bielsko-Biala) on a commonly available plastic board for electronic usage, covered by an 0.35 μm copper layer. After removal of some thermo-transfer paper, the printed-boards were cleaned and placed in a chemical, a solution liquid aiming for removal of copper from the areas which were not covered by paths indicated by the toner patterns. In the next manufacturing step, the

Fig. 1 Chassis of DUODEPED (3D CAD model)



paths remaining on the printed board were covered by leadless stannum and preserved by means of a ‘soldermask’ varnish. Finally, electronic elements were soldered;

- (iii) the next phase was divided in the following steps:
- writing a program for the micro-controller **PIC24HJ256GP610** made by Microchip,
 - tuning of communication modules Bluetooth BTM222,
 - writing an application for controlling the robot—for the mobile device (mobile phone) based upon Android operational system,
 - performance of visualization SCADA system for controlling of the robot by means of PC-type computer or laptop using ‘inTouch’ software based upon the ‘Windows’ operational system,
 - performance of robot vision system based upon the computer ‘**Raspberry PI**’ platform and the camera of 5MP resolution.

The final result of all performed design and manufacturing activities is the robot presented in Fig. 2.

Underneath some additional considerations concerning the general layout, motion and robot control system are added. The robot side view as well as control possibilities are given in Fig. 3—e.g. phase shifts of motions of the row of three legs on the right side of the robot as well as the skeleton of the leg on the left side of the robot, respectively.

The motion system of a robot consists of 12 legs which ensure legs drive and legs movement as well as stability of the robot itself. There are 12 legs arranged in 4 triples of legs i.e. 2 triples on both sides of the robot. In each triple, always a single leg has contact with the ground. The continuous activities of sets of three legs insure that each leg of the set has contact with the ground for 1/3 of one full rotation of the driven crankshaft.

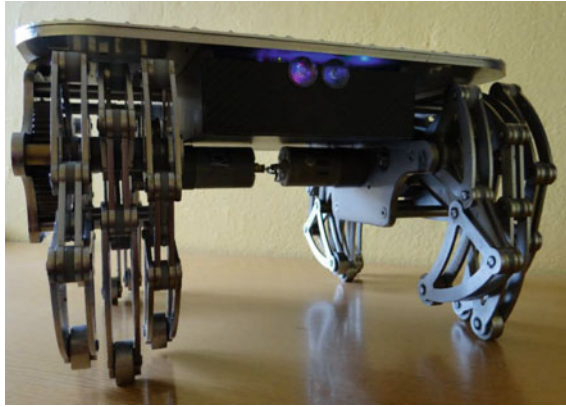


Fig. 2 Assembled walking robot

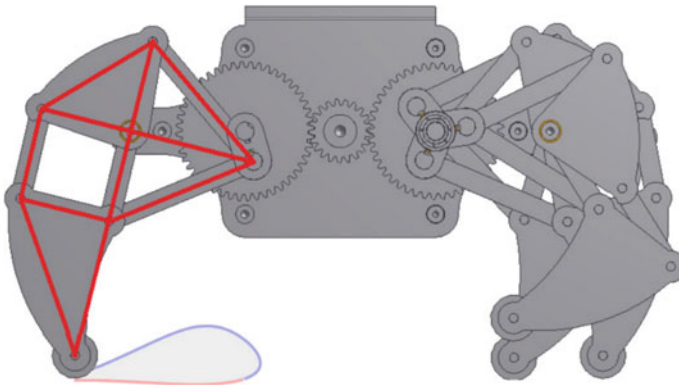


Fig. 3 Robot—side view and control possibilities

The legs are driven in two pairs on two robot sides as well as in pairs: front leg and rear leg (one by one) in an adequate motion routine. Therefore the pair of legs, front leg and adequate rear leg, are coupled via the drive system. The mechanism is constructed by a sub-system of geared wheels and crankshafts mounted on geared wheel axes as well as design solutions of the legs themselves. The crankshafts are shaped in such a manner as to accomplish phase-shifts by the special delay angles. The single leg itself is built as a mechanism consisting of several kinematical pairs. The rear and front legs are the same but positioned in mirror symmetry. The 3D view of two legs is presented in Fig. 4. One can observe geared wheels in contact, the leg joints and leg links as well as the positions of crankshafts. The detailed front view of a single leg is presented in Fig. 5. The small wheel is mounted on the leg end (Figs. 4, 5 and 6). In fact, each leg consists of approx. 40 elements. We have simplified the considered model by omitting bearings and replacing parallel

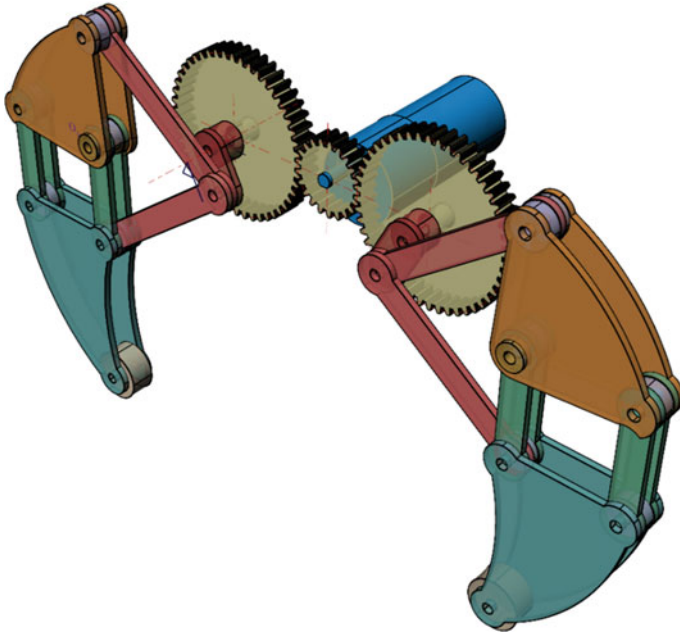


Fig. 4 Single pair of legs (from opposite triple) and their driving motor

linkages by planar elements. Between the moving elements, the Teflon washers (0.5 mm thickness) were placed to diminish friction on co-operating links' surfaces. Aiming for reduction of weight and dimensions, instead of ball bearings, slide bearings have been utilized. The slide bearing consists of a brass sleeve rotating on a shaft.

Aiming for preservation of stability, the mounting points of the legs were placed in points that achieve the separation distance of 194 mm. The planar mechanism as a leg representation is analyzed for simplicity of considerations. The chosen dimensions of leg elements are given in Figs. 5 and 6. The legs were made of special steel AISI 304L—1.4306 (according to the standard PN 00H18N10). The leg structure is created from triangles and beam-type links. The legs are driven by an electric motor shown in the center in Figs. 4, 5 and 6. The rotational movement is transmitted onto legs via the system of geared wheels (w_1 and w_2) and crankshafts. The beams (1A and 1B) are connected to the end axes of the crankshafts as well as to triangle leg parts (2 and 4). The beams (rods) (3A and 3B) complete a construction that enables robot walking activities.

In the next chapter, a graph model of one robot leg is considered. We utilized bond graphs due to their generality, common knowledge and availability of free open-source software.

Since, the considered system consists of plenty of elements, which realize rotational and planar motion, the nontraditional method is needed. There are a few methods of planar mechanisms analysis. The graphical method (for example the

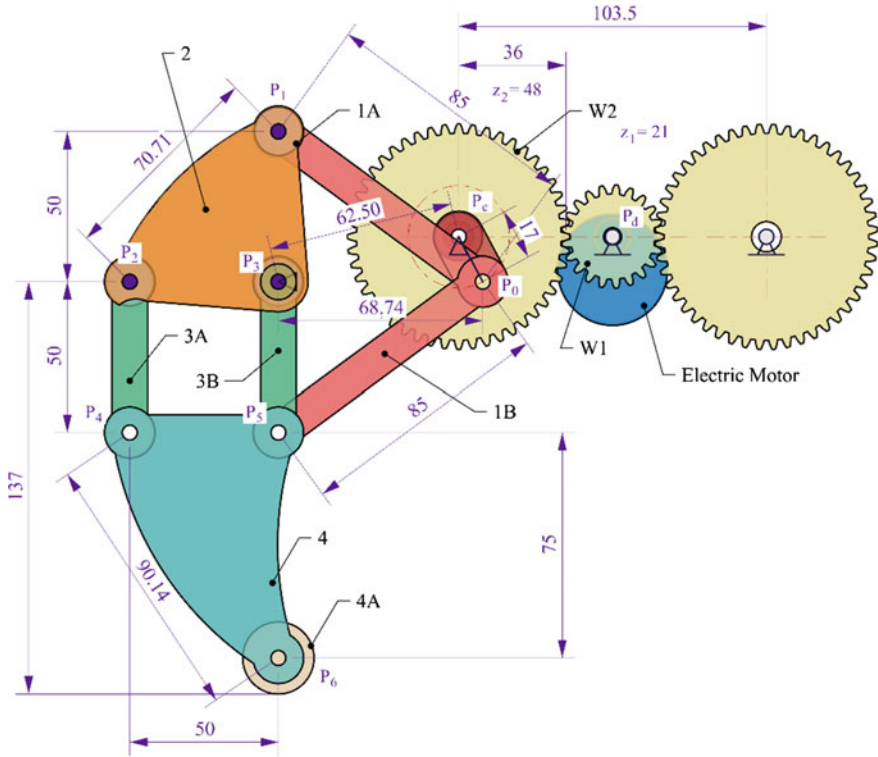


Fig. 5 Front view of driving system, for clarity only one leg (from three in the row) was shown

velocity and acceleration polygon method) has been used for a long time. They give consistent and obvious solutions. The algebraic method based on Newton’s equations and Euler’s formulae, and they are more useful for more complicated devices. They need also dedicated computer software or routines.

The alternative approach is to utilise the methods based on a unified system representation, i.e. a unified set of variables. These variables are grouped in pairs (effort and flow variable: $e f$ respectively): force and linear displacement, moment of force and angular velocity, volume flow and pressure, and others. One can notice [8, 9], that the product of these variables is always power, and we can write down the general formula:

$$P = e \cdot f. \tag{1}$$

So, we have tools for describing whole multi-domain systems. Bond graphs are based on these considerations and assumptions.

3.1 Graph for a Single Body

A model of a single solid body (the simple linkage) consists of a set of 1-nodes and 0-nodes, additionally the modulated transformers MTF were added for describing the nonlinear transfer of energy between basement (ground) and moving body. The weights (modules) of modulated transformer have matrix form, i.e.:

$$\mathbf{T}_i = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix}_i \quad (3)$$

where:

c_{ij} coefficient of transformation, i.e. rotation matrices, and relationship in local and global coordinates [24].

The bond graph for a single body is shown in Fig. 7a, b. This is the simplest linkage with two joints. We assume that energy flows only through these joints: O_1 and O_2 . This model will be used for elements 1A, 1B, 3A, 3B (see Figs. 5 and 6).

Additionally, a bond graph for 3-joint linkage was built for modeling elements 2 and 4 (Figs. 6 and 7). There are 3 joints by means of which energy flows through the subsystem; the bond graph for this body was depicted in Fig. 7c, d.

Using these two basic bond graph models and connecting them, we can obtain a model for a whole robot leg. The bond graph for a single kinematic chain (single leg) is depicted in Fig. 8. The robot has twelve legs, however for simplicity, we consider only one.

The graph model we have obtained consists of five bond graphs modeling every linkage of leg mechanism. They are connected with the auxiliary bonds depicted in Fig. 7.

This model requires excitation, so we put in a few source elements. The driving force from electric motor is equal to

$$F_G = \frac{i_{12}}{l_C} M_{mo} = \left(\frac{z_2}{z_1} \right) \frac{1}{|P_C P_O|} M_{mo} \quad (4)$$

where:

z_1, z_2 numbers of teeth of geared wheels,

l_C crank length,

M_{mo} torque of electric motor.

The effort source \mathbf{Se} : F_G describes this excitation. Additionally the gravity forces $m_i g$ acting on structure elements are included in the model with effort sources \mathbf{Se} : $[0 \ m_i g]^T$.

Inertial forces acting on the system are included in the model by means of generalized inertia elements I . Since the planar motion is inherent, there are two kinds of these elements:

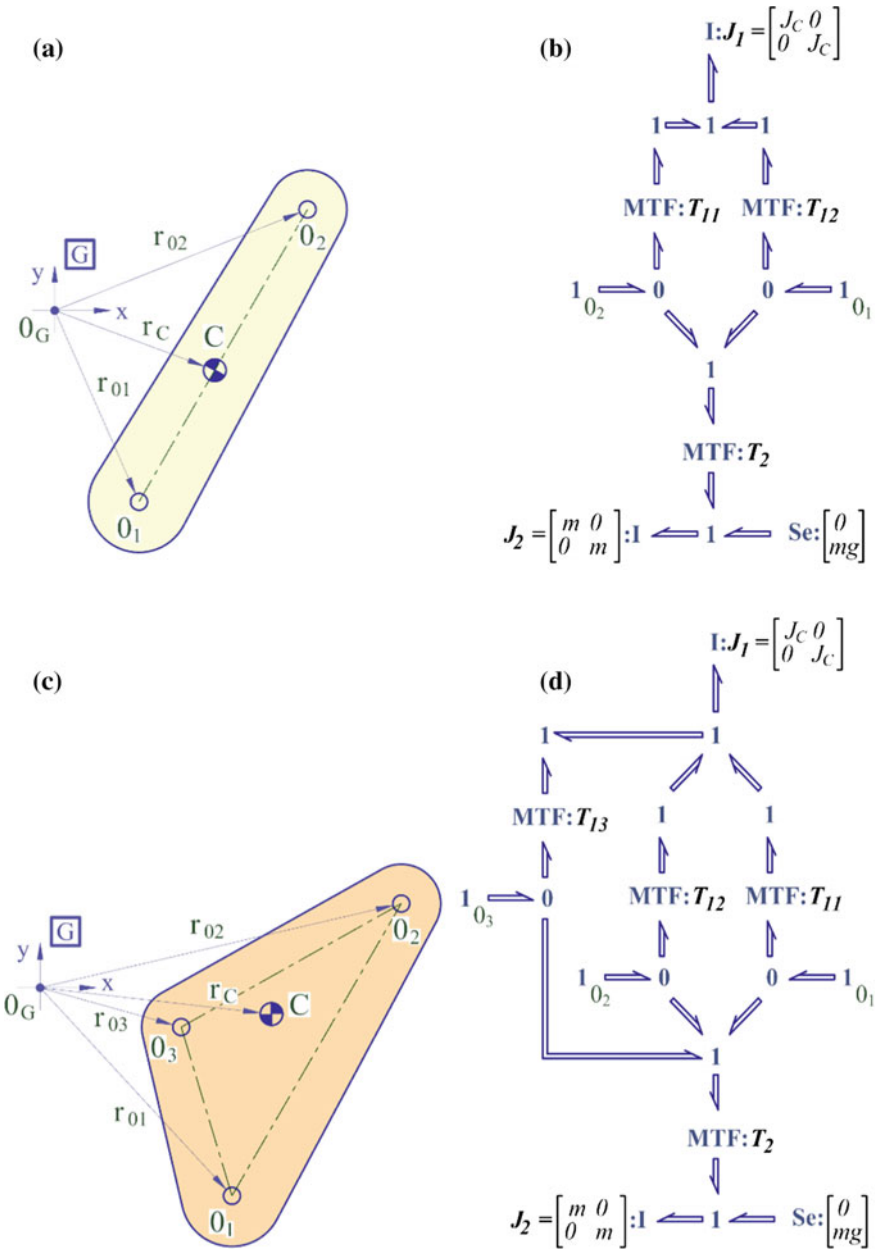


Fig. 7 Simple linkages and their bond graphs: (a) and (b): for a two joint element, (c) and (d): for a three joint element, respectively

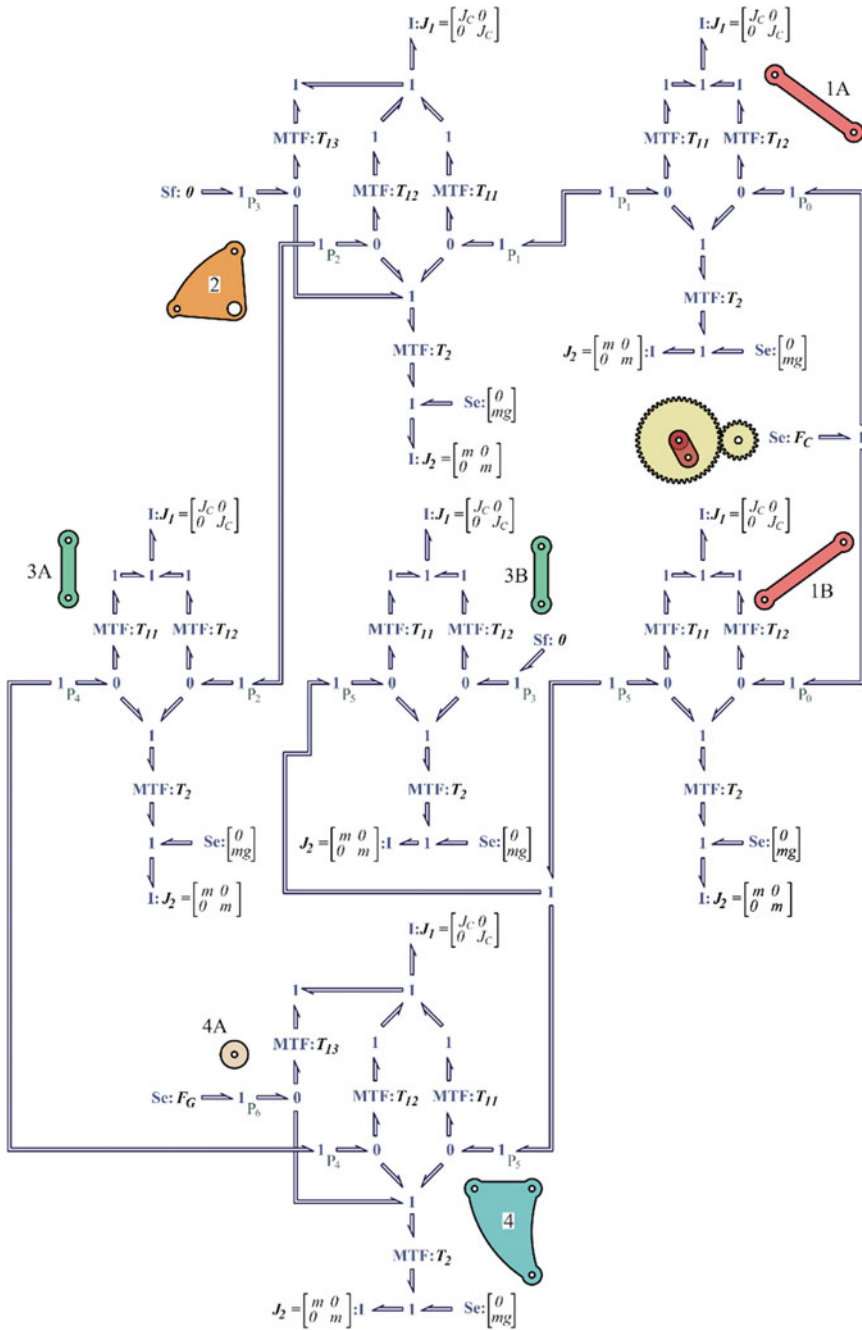


Fig. 8 Single robot leg—bond graph model of the leg

$$I : \begin{bmatrix} m_i & 0 \\ 0 & m_i \end{bmatrix} \quad (5)$$

inertial forces of linear motion and

$$I : \begin{bmatrix} J_{Ci} & 0 \\ 0 & J_{Ci} \end{bmatrix} \quad (6)$$

inertial forces of rotational motion with respect to the mass center of the elements. The indices in Fig. 7 were omitted, and the reader should trace numbers of elements and their pictograms. The completed model can be used for simulation of robot movement, utilizing commercial software or dedicated software.

4 Conclusions

In this paper, we describe bond graph models of robot legs. The robot was built and tested in various conditions. The bond graph models allow for performance of simulation and testing of some design conditions and parameters. The obtained results confirm the usefulness of bond graph modeling as a universal and effective way of testing in the phase of design or redesign of robots.

Further analyses and performing simulation by means of the created bond graph model will be a subject of further works of the authors' team.

References

1. Gola, B.: Robot, (in Polish). Diploma thesis, University of Bielsko-Biala, Bielsko-Biala, Poland (2014)
2. Gor, M.M., et al.: Dynamic modeling and simulation of compliant legged quadruped robot. In: Proceedings the 1st International and 16th National Conference on Machines and Mechanisms, IIT Roorkee, India, pp. 7–15 (2013)
3. Kayani, S.A., Malik, M.A.: Modeling and simulation of biped kinematics using bond-graphs. In: ICET'06 International Conference on Emerging Technologies, pp. 677–682 (206)
4. Drewniak, J., Zawiślak, S.: Linear graph and contour graph-based models of planetary gears. JTAM **48**(2), 415–433 (2010)
5. Dudek, G., et al.: Robotic exploration as graph construction. IEEE Trans. Robot. Autom. **7** (19), 589–865 (2013)
6. Rekleits, I.M., Dudek, G., Milos, E.E.: Graph-based exploration using multiple robots. Distrib. Auton. Robot. Syst. **4**, 241–250 (2000)
7. Wojcik, L.: Modeling of musculoskeletal structure and function using a modular graph approach. J. Franklin Inst. **340**, 63–76 (2003)
8. Borutzky, W.: Bond Graph Methodology: Development and Analysis of Multidisciplinary Dynamic System Models. Springer (2009)

9. Karnopp, D.C., Margolis, D.L., Rosenberg, R.C.: *System Dynamics Modeling Simulation and Control of Mechatronic Systems*, 5th edn. Wiley (2012)
10. Zawislak, S.: *The graph-based methodology as an artificial intelligence aid for mechanical engineering design*. Habilitation Thesis, University of Bielsko-Biala, Poland (2010)
11. Bailey, T., et al.: Data association for mobile robot navigation: a graph theoretic approach. *Proceedings of IEEE International Conference on Robotics and Automation* **3**, 2512–2517 (2000)
12. Tazaki, Y., Imura, J.I.: Graph-based model predictive control of a planar bipedal robot. *J. Robot. Soc. Japan* **24**(5), 101 (2006)
13. Yang, G., Chen, I., Lin, W., Angeles, J.: Singularity analysis of three-legged parallel robots based on passive-joint velocities. *IEEE Trans. Robot. Autom.* **17**(4), 413–422 (2001)
14. Yao, G., Chen, I.M.: Task-based optimization of modular robot configurations: minimized degree-of-freedom approach. *Mech. Mach. Theory* **35**, 517–540 (2000)
15. Seth, A.: *A predictive control method for human upper-limb motion: graph-theoretic modelling, dynamic optimization, and experimental investigations*. Master of Science Thesis, University of Waterloo, Waterloo, Canada (2000)
16. Wojnarowski, J., Kopec, J., Zawislak, S.: Graphs and gears. *JTAM* **1**(44), 139–162 (2006)
17. Zawislak, S., et al.: Some applications of graph transformations for modeling of mechanical systems. In: *Proceedings of the International Workshop on Graph Transformations and Visual Technique*, Budapest, pp. 332–345 (2008)
18. Chen, I.-M., Yang, G.: Configuration independent kinematics for modular robot. In: *Proceedings of IEEE International Conference on Robotics and Automation*, 2, 1440–1445 (1996)
19. Chen, I.-M., Yeo, S.H., Yang, G.: Kernel modular robot applications: automatic modeling technique. *Int. J. Rob. Res.* **18**(2), 225–242 (1996)
20. Gal, A., Vladereanu, L.: Sliding mode control with bond graph modeling applied on a robot leg. In: *Proceedings the 2014 International Conference on Circuits, Systems and Control*, pp. 40–50 (2014)
21. Ghafari, A.S., Meghdari, A., Vossoughi, G.R.: Modeling of human lower extremity musculo-skeletal structure using bond graph approach. *ASME International Mechanical Engineering Congress and Exhibition*, pp. 1489–1498 (2007)
22. Mohanty, P.K., Parhi, D.R.: Controlling the motion of an autonomous mobile robot using various techniques: a review. *J. Adv. Mech. Eng.* **1**, 24–39 (2013)
23. Sandoval-Castro, X.Y., et al.: Kinematics of Hex-Piderix—A six legged robot—Using screw theory. *Int. J. Adv. Robot. Syst.* **10**(19) (2013)
24. Ragusila, V., Emami, M.R.: Modelling of a robotic leg using bond graph. *Simul. Model. Pract. Theory* **40**, 132–143 (2014)

Kinematical Analysis of Variants of Wind Turbine Drive by Means of Graphs

J. Drewniak, J. Kopeć and S. Zawiślak

Abstract Variants of a wind turbine drive are discussed in the paper. The drive should be adjustable. The control tools should be reliable and simple. Several variants of drive system based upon a compound planetary gear accompanied by a friction gear are analysed and compared. Graph models had been used for analysis of the gear itself. The analysis of the whole system using graphs is proposed via modification of known approaches. The analysis of results shows their full compatibility.

Keywords Drive of wind power station · Mixed multi-graph methods · Modified contour graphs

1 Introduction

Drive mechanisms of wind turbines—due to the variations of wind directions and power—should be controlled in an adequate way to assure constant angular velocity of a generator. In the present paper, the chosen drive subsystems have been described, which could be recognized as mechanically flexible transmissions.

The analysis of these systems can be done by means of graph modeling. The advantages of graph methods are: general approach, possibility of algorithmization and easiness of modification.

In the present paper, we propose modified graph approaches, besides the traditional approach that allows for comparison of results and gives deeper insight into the performed analyses.

Recently, bond graphs were applied for solving versatile problems of modeling of wind turbines e.g.: modeling of blades [1], detection of damages [2] or modeling and simulation of the whole drive system of wind turbine [3, 4, 6, 14–19, 21]. The

J. Drewniak (✉) · J. Kopeć · S. Zawiślak
Faculty of Mechanical Engineering and Computer Science,
University of Bielsko-Biała, Bielsko-Biała, Poland
e-mail: jdrewniak@ath.bielsko.pl

novelty of the modified graph approach consists in extension of the known graph models on systems with coupling elements (i.e. friction transmissions) which till now has not been considered too frequently. In the paper [11], the planetary systems coupled with various types of coupling are described e.g. the continuously variable belt transmission has been utilized, whereas modeling was performed using so-called hypergraphs. However, these structures are different from the classic hypergraphs used in the paper [18].

2 Kinematical Analysis of Planetary Gear Systems

In the present paper, the kinematical analysis of the considered gears have been performed via traditional as well as graph methods [7–9, 19, 21]. Firstly, we consider planetary gear presented in Fig. 1. The gear system should be adjustable—allowing achieving of adequate rotational velocity of the generator’s shaft. The system should work as a multiplier. As a first step, the simple functional scheme was proposed. The elements of the gear are as follows: carriers h_1 and h_2 , planetary wheels 2 and 5, sun wheels 1 and 4, as well as ring gears 3 and 6.

Mobility of the considered gear (Fig. 1) is equal:

$$W = 3 \times n - 2 \times p_5 - p_4 = 3 \times 6 - 2 \times 6 - 4 = 2, \quad (1)$$

where: n —number of moving bodies, p_5 and p_4 —number of one degree of freedom joints and number of two degrees of freedom joints.

Therefore the system consisting of the planetary gear accompanied by continuously variable friction transmission is proposed. The variants are given in Figs. 2, 3 and 4.

The considered numbers of teeth are given in Table 1, showing also so-called Willis notation. In case of graph methods all teeth numbers relevant to a ring gear are considered as positive, therefore the reader should pay attention to the convention of notation.

After an analysis of the afore-mentioned variants, we propose the resultant design solution i.e. planetary gear with coupling of chosen elements—Figs. 5 and 6. Mobility for the final design solution of the considered gears has changed:

$$W = 3 \times n - 2 \times p_5 - p_4 = 3 \times 7 - 2 \times 7 - 6 = 1. \quad (2)$$

The modified structures (Figs. 5 and 6) allow not only for reduction of DOF to one but also allow for increasing of angular velocity via changes of friction transmission parameters (r_7, r_8). Some analyses of possible variants of the system are given underneath for chosen teeth numbers.

These are the exemplary data which give an overview of the possible ranges and sizes. The final design and its parameters have to be fitted to the detailed site, current generator and wind conditions which is beyond the scope of the present paper.

Fig. 1 Functional scheme of the gear applied in the wind turbine drive

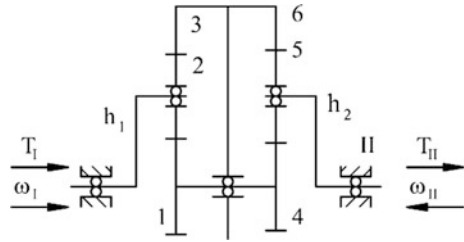


Fig. 2 First variant of wind turbine planetary gear-friction gear set with free carrier h_1

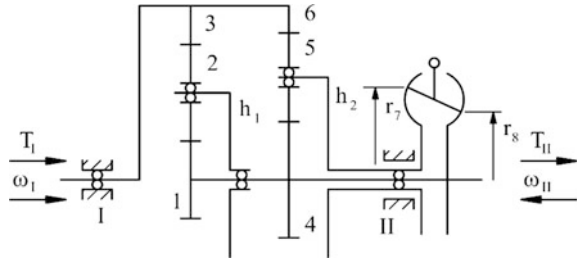


Fig. 3 Second variant of wind turbine planetary gear-friction gear set with free ring gear

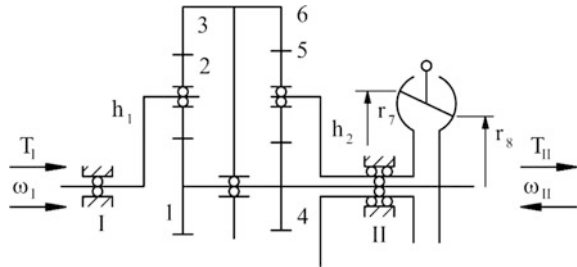


Fig. 4 Third variants of wind turbine planetary gear-friction gear set with free carrier h_2

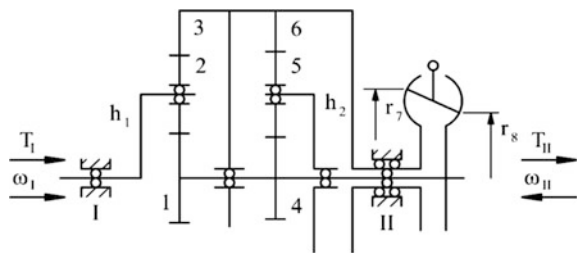


Table 1 Numbers of teeth^a

I row	II row
$z_1 = 19$	$z_4 = 19$
$z_2 = 50$	$z_5 = 44$
$z_3 = 119$	$z_6 = 107$

^aRemark. According to notation of the Willis's method: $z_3 = -119$ and $z_6 = -107$

Fig. 5 Simplified variant of the gear; input on ring gear

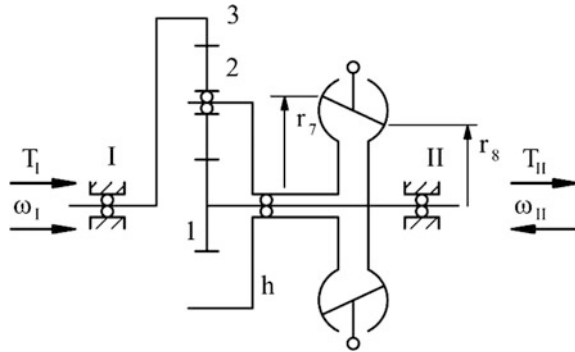
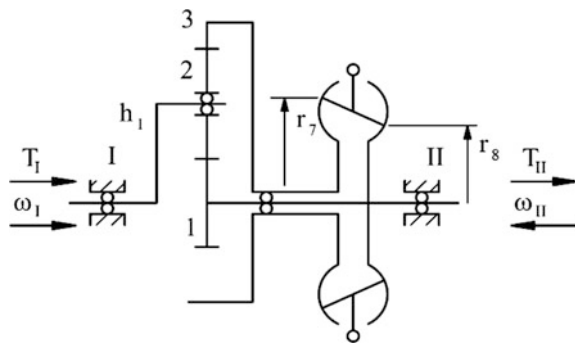


Fig. 6 Simplified variant of the gear; input on carrier



2.1 Classical Analysis of Drive Variants

The system with a driven ring gear (input angular velocity ω_3 , Fig. 2) is considered as the first one. Carrier h_1 is a free part in this case. It is not engaged besides carrying planetary wheels, therefore Eq. (3) will not be utilized to obtain the solution. Based on the schema of the gear set we have: $\omega_6 = \omega_3 = \omega_I$ —known value and $\omega_4 = \omega_1 = \omega_{II}$ —unknown value, respectively. Additionally, ω_{h1} is self-adjusted depending on ω_3 and ω_1 , where the following notations are utilized: ω_i —angular velocity of element number i , z_j —number of teeth on the geared wheel j .

Based on Willis approach, we can write down the equations:

$$i_{3,1}^{h1} = \frac{\omega_3 - \omega_{h1}}{\omega_1 - \omega_{h1}} = \left(\frac{z_1}{z_3} \right), \tag{3}$$

$$i_{6,4}^{h2} = \frac{\omega_6 - \omega_{h2}}{\omega_4 - \omega_{h2}} = \left(\frac{z_4}{z_6} \right) \tag{4}$$

where $i_{3,1}^{h1}$ is the kinematical ratio for the first row wheels (2, 3, 1) with respect to carrier h_1 . Similarly, $i_{6,4}^{h2}$ is the kinematical ratio for the second row wheels (6, 5, 4) with respect to carrier h_2 . The kinematical ratio of the friction transmission is equal:

$$i_f = \frac{\omega_{h2}}{\omega_4} = -\frac{r_8}{r_7(1-\varepsilon)}, \quad (5)$$

where: r_7, r_8 —radii defined by distances between axis and contact points,
 ε —friction coefficient (in this case it could be omitted).

From Eqs. (4) and (5), the formula for output angular velocity ω_{II} was obtained as well as the total ratio of gear set:

$$i_T = \frac{\omega_I}{\omega_{II}} = \frac{\omega_6}{\omega_4} = \frac{z_4}{z_6} + i_f \left(1 - \frac{z_4}{z_6}\right). \quad (6)$$

Taking into account numerical value for $i_f = -0.5$, the value of the ratio is equal:

$$i_T = \frac{19}{-107} + (-0.5) \left(1 - \frac{19}{-107}\right) = -0.766. \quad (7)$$

Absolute value of the total gear set ratio $|i_T| < 1$, therefore the gear set presented in Fig. 2 is a multiplier. Based on these considerations, this design solution has been approved as possible for practical application.

An analysis of the 2nd variant is performed in what follows (Fig. 3). The assumptions are: the driven element is carrier h_1 , whereas ring gear 3, 6 connects two rows of gears, only. Willis's equations are the same as Eqs. (3) and (4), similarly the kinematical ratio of friction transmission is the same as in the previous case, see (5).

Based on Fig. 3 one can find: $\omega_6 = \omega_3$, $\omega_{h1} = \omega_I$ (known quantity), $\omega_I = \omega_4 = \omega_{II}$.

After calculations:

$$i_T = \frac{\omega_I}{\omega_{II}} = \frac{\omega_{h1}}{\omega_4} = \frac{i_f \left(1 - \frac{z_4}{z_6}\right) - \left(\frac{z_1}{z_3} - \frac{z_4}{z_6}\right)}{1 - \frac{z_1}{z_3}}. \quad (8)$$

Taking into account $i_f = -r_8/r_7r_7$, some exemplary results are shown in Table 2.

The absolute value of gear set $|i_T| < 1$.

Table 2 Numerical calculations for formula (8) for different i_f values

i_f	i_T	$1/i_T$
-0.5	-0.523	-1.911
-0.3	-0.320	-3.124
-0.2	-0.219	-4.576

The third variant (Fig. 4) seems to be the best of three analyzed ones, because the following formulas (derived from the Willis method) hold:

$$\omega_1 = \omega_H = \omega_{h1} \frac{1 - \frac{z_1}{z_3}}{i_f - \frac{z_1}{z_3}}. \quad (9)$$

Total ratio of gear set is (for $i_f = -0.5$) is:

$$i_T = \frac{i_f - \frac{z_1}{z_3}}{1 - \frac{z_1}{z_3}} = \frac{-0.5 - \frac{19}{-119}}{1 - \frac{19}{-119}} = -0.294. \quad (10)$$

Values of total ratios and their reciprocals are collected in Table 3.

The absolute value of gear set ratio are lower than 1, as previously: $|i_T| < 1$.

Since carriers h_1 and h_2 in design solutions presented in Figs. 2 and 4 respectively have no any influence on kinematics of the system, they could be replaced with a simple variant i.e.: the simple planetary gears coupled with the friction transmission shown in Figs. 5 and 6, respectively. As it was mentioned, analogical proposals are given in [10] but modeling is performed in another way. The kinematics of the discussed system could be adjusted to change the configuration of the engaged elements of the friction transmission.

Finally, an analysis of the gear system (presented in Fig. 5) is performed. The simplified system is analyzed as the possible optional version taking into account the fact that free rotating elements could be omitted. The Willis formula takes the following form:

$$\frac{\omega_3 - \omega_h}{\omega_1 - \omega_h} = \left(\frac{z_1}{z_3} \right). \quad (11)$$

Due to the fact that elements h and 1 are connected via friction gear, the relationship between adequate rotational velocities can be written:

$$i_{h,1} = \frac{\omega_h}{\omega_1}. \quad (12)$$

Using formula (12) we obtain:

$$\omega_3 - \omega_1 \cdot i_{h,1} = \left(\frac{z_1}{z_3} \right) \cdot \omega_1 - \left(\frac{z_1}{z_3} \right) i_{h,1} \cdot \omega_1. \quad (13)$$

Table 3 Additional calculations for formula (10) for different i_f values

i_f	i_T	$1/i_T$
-0.5	-0.2935	-3.407
-0.3	-0.1210	-8.263
-0.2	-0.0348	-28.750

After some recalculations we have:

$$\omega_1 \cdot \left(i_{h,1} + \frac{z_1}{z_3} - \frac{z_1}{z_3} i_{h,1} \right) = \omega_3. \tag{14}$$

For $i_{h,1} = -2$, the ratio is equal:

$$i_{3,1} = \frac{\omega_3}{\omega_1} = i_{h,1} + \frac{z_1}{z_3} - \frac{z_1}{z_3} i_{h,1} = -2.48. \tag{15}$$

The simplified gear system, presented in Fig. 6, can be analyzed in the same way. We can summarize the calculations concluding that an addition of a friction gear allows for changing the ratio. Moreover, together with some number of teeth changes, it allows for creation of family of variants. It gives a designer a possibility of choice of the most suitable design solution.

3 Graph-Based Analysis of Planetary Gear Sets

3.1 Mixed Multi-graphs Models

The similar analyses can be performed by versatile modeling methods e.g. graph-based approach; using mixed, contour or bond graphs [15]. The mixed graphs for the gear schemes shown in Figs. 2, 3, 4 and 5 are presented in Fig. 7a–d, respectively.

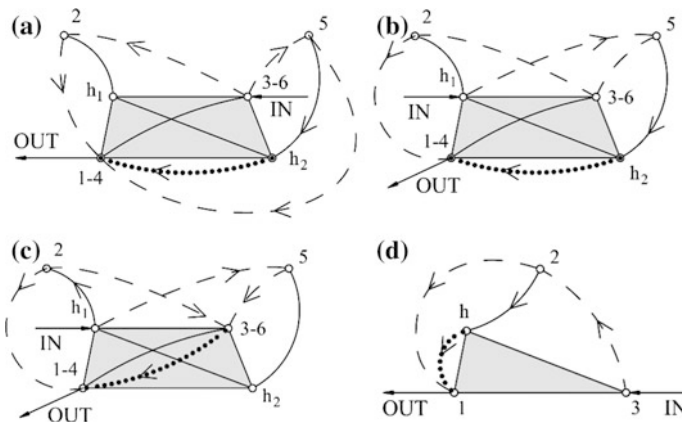


Fig. 7 Mixed multi-graphs representing the gear systems: **a–c** equivalent to schemes in Figs. 2, 3 and 4, respectively; **d** representing the simplified design in Fig. 5

The applied graphs are the modified mixed graphs discussed by authors in their previous publications [7–9, 21]. Here additionally, a new type of edges is introduced i.e. an edge drawn by means of a dotted line that symbolically represents the coupling among the elements connected via a friction clutch. Therefore the considered graphs are in fact mixed multi-graphs because some pairs of vertices are connected by means of two parallel edges or arcs, simultaneously.

Analyzing Fig. 7a, one can see that vertex 3–6 represents an input element (indicated by an arrow *IN*), vertex 1–4 represents an output element (indicated by an arrow *OUT*). Directed paths from input to output show passing of rotational movement onto consecutive gear elements. The edge $(2, h_1)$ was not converted into an arc which indicates that carrier h_1 is a free element. Vertices 1–4 and h_2 are connected by an edge and an arc simultaneously. The edge means that both elements rotate around main geometrical axis of the gear system whereas the dotted arc symbolizes coupling of these two elements. Analysis of directed paths, coupling edges, dashed edges etc.—gives an insight into the work duties of the considered gear. Till now, such modified graphs have not been considered as models of any mechanical systems. Obviously, the introductory rules of modeling are known, however they are not widely used.

The method consists in finding so-called f-cycles [20] which are listed via their codes (16). The code represents an elementary subsystem of two gear wheels and carrier. Based upon these codes the system of kinematical equations can be written in an algorithmic way.

$$(1, 2)h_1 \quad (2, 3)h_1 \quad (4, 5)h_2 \quad (5, 6)h_2. \quad (16)$$

The code $(i, j)k$ denotes gearing pair symbols (in brackets) and description of carrier—outside the brackets. The pair in brackets should be ordered from a lower to a higher description. The rules of coding allow for algorithmic creation of kinematical equations, especially order and indices. The derived system of equations is shown in a system of Equation (17). Additional equations describe the design special details i.e. common rotational velocities of geared wheels fixed on the same shaft or ring gear. Moreover, the very end equation represents the dotted line edge. The input and output to the system is represented in the discussed graphs by means of external arrows indicating special vertices of the graph.

Additionally, the path or set of paths are drawn showing passing of rotational movement and power throughout the gear from the input to the output. We consider directed paths from the input to the output, therefore the adequate edges are converted into arcs. The edges belonging to a polygon—its boundary edges as well as diagonals are always considered as edges. The edges outside the polygon which are not turned into arcs represent free moving elements or temporary redundant elements when the design solution allows for different modes of operation (e.g. for automotive gear boxes).

The signs plus or minus in system (17) depend on internal or external meshing of gear elements, respectively. The complete system of equations is as follows:

$$\left\{ \begin{array}{l} \omega_1 - \omega_{h1} = -N_{21}(\omega_2 - \omega_{h1}) \\ \omega_2 - \omega_{h1} = +N_{32}(\omega_3 - \omega_{h1}) \\ \omega_4 - \omega_{h2} = -N_{54}(\omega_5 - \omega_{h2}) \\ \omega_5 - \omega_{h2} = +N_{65}(\omega_6 - \omega_{h2}) \\ \omega_1 = \omega_4 \\ \omega_3 = \omega_6 \\ \frac{\omega_{h2}}{\omega_4} = i_f \end{array} \right. \quad (17)$$

Based upon this system, the rotational velocity of the velocity of an output element can be obtained knowing the input data. The ratio of the gear for the first variant (Figs. 2 and 7a) is as follows:

$$\omega_4(1 - i_f - i_f \cdot N_{65} \cdot N_{54}) = -N_{65} \cdot N_{54} \cdot \omega_6, \quad (18)$$

$$\frac{\omega_6}{\omega_4} = -\frac{1 + i_f(-1 - N_{65} \cdot N_{54})}{-N_{65} \cdot N_{54}}. \quad (19)$$

The solution (19) can be rewritten for the assumed numbers of teeth of the adequate wheels:

$$i_T = \frac{\omega_6}{\omega_4} = -\left[\frac{z_4}{z_6} + i_f \left(-\frac{z_4}{z_6} - 1 \right) \right] = -\left[\frac{19}{107} + (-0.5) \left(-\frac{19}{107} - 1 \right) \right] = -0.766 \quad (20)$$

The same result as in formula (7) was obtained. As it can be seen, the first two equations of the system were not utilized in the solution at all because the carrier h_1 is a free rotating element.

For simplified system (Fig. 5), the equations of motion are as follows:

$$\left\{ \begin{array}{l} \omega_1 - \omega_{h1} = -N_{21}(\omega_2 - \omega_{h1}) \\ \omega_2 - \omega_{h1} = +N_{32}(\omega_3 - \omega_{h1}) \\ \frac{\omega_h}{\omega_1} = i_{h,1} \end{array} \right. \quad (21)$$

The mixed multi-graph model is presented in Fig. 7d. The input is on the element 3 and the output on the element 1, respectively. The solution of the system of equations (assuming as previously $i_{h,1} = -2$) is as follows:

$$\frac{\omega_3}{\omega_1} = -\frac{-1 + i_{h,1}(1 + N_{32}N_{21})}{N_{32}N_{21}} = -\frac{z_1}{z_3} + i_{h,1} \left(1 + \frac{z_1}{z_3} \right) = -2.48. \quad (22)$$

We can conclude that, as previously, the results of Willis and mixed graph methods are fully compatible.

3.2 Modified Contour Graph Method

Additionally, the simplified system was analyzed by means of contour graph method [9]. The contour for the chosen layouts of the considered systems are presented in Fig. 8. The Marghitu's conventions for graph drawing and naming cycles as contours were preserved. We consider here only the simplified solution discussed in Figs. 5 and 7d. The adequate contour graph is given in Fig. 8c. In the considered case, the following codes of contours are distinguished: $(0, 3, 2, h, 0)$ and $(0, 3, 2, 1, 0)$. As in the previous case, the codes allow for algorithmic derivation of equation system (23).

The system consists of vector equations, every contour generates two equations. The solution of the system (23) consists in turning it into the system of scalar equations entering the trigonometric functions and adequate values of angles between the vectors. The angle between the radius and rotational velocity (treated as vectors) is equal to 90° and all vectors of the equations first and third are

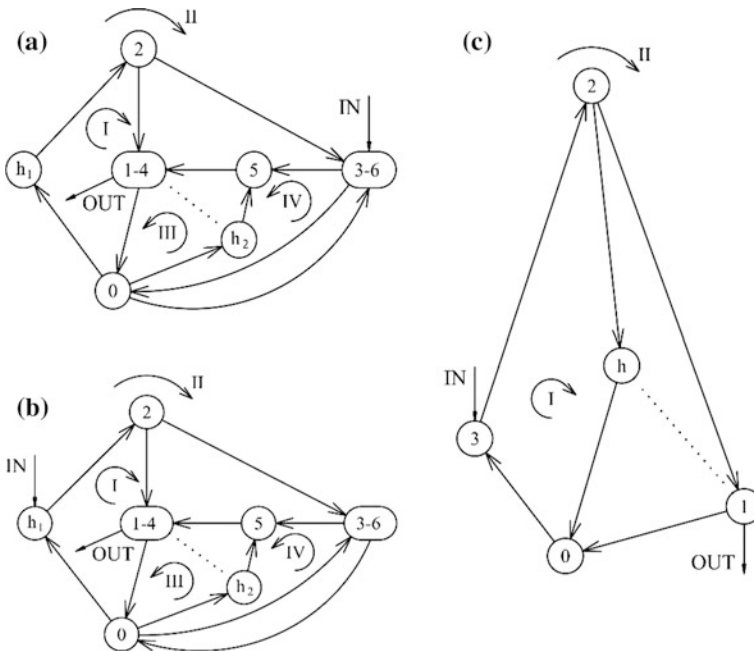


Fig. 8 Contour graphs of the gears: **a** and **b** equivalent to schemes in Figs. 2 and 3, respectively; **c** representing the simplified design in Fig. 5

uni-directional. It allows for conversion of the vector equations into the scalar ones in a very simple manner.

$$\begin{cases} \boldsymbol{\omega}_{3,0} + \boldsymbol{\omega}_{2,3} + \boldsymbol{\omega}_{1,2} + \boldsymbol{\omega}_{0,1} = \mathbf{0} \\ \mathbf{r}_{2,3} \times \boldsymbol{\omega}_{2,3} + \mathbf{r}_{1,2} \times \boldsymbol{\omega}_{1,2} = \mathbf{0} \\ \boldsymbol{\omega}_{3,0} + \boldsymbol{\omega}_{2,3} + \boldsymbol{\omega}_{h,2} + \boldsymbol{\omega}_{0,h} = \mathbf{0} \\ \mathbf{r}_{2,3} \times \boldsymbol{\omega}_{2,3} + \mathbf{r}_{h,2} \times \boldsymbol{\omega}_{h,2} = \mathbf{0} \end{cases} \quad (23)$$

where:

$\omega_{i,j}$ —relative rotational velocity i th element in relation to j th element.

$r_{i,j}$ —radius measured from the main gear geometrical axis to the adequate point e.g. co-operation of gear element; if i and j denote sun wheel and planet then the common point of their pitch diameters is considered. The radius can be expressed by means of modulus and teeth numbers. The same modules are assumed. They are canceled, therefore only teeth numbers are present in the final formula (25).

The solution of the system is the same as previously. Additionally, the solution consists in conversion of relative rotational velocities into the absolute ones (in relation to the ground, support of the gear system). An additional equation is simultaneously considered:

$$i_{h,1} = \frac{\omega_h}{\omega_1}. \quad (24)$$

The numerical solution (for $i_{h,1} = -2$) is as follows:

$$i_{3,1} = \frac{\omega_3}{\omega_1} = \frac{z_1 - 2 \cdot (z_1 + z_2) \cdot i_{h,1}}{-z_1 - 2z_2} = \frac{295}{-119} = -2.48. \quad (25)$$

In case of contour graphs for the simplified system (Figs. 5, 7d and 8c), the same result was obtained as previously, for Willis and mixed multi-graph methods. It confirms the consideration as well as shows that generalization of the graph methods for the gears with coupling elements has been successfully proposed and performed.

3.3 Bond Graphs Models

Bond graphs are used frequently for modelling of versatile wind drive mechanical and mechatronic systems [1–4, 14–17, 20, 22]. The well-known monographs dedicated to theory and application of the bond graphs are e.g. books [5, 15]. Planetary gears are modeled by means of bond graphs in [10] which is very rare.

The considered bond graphs (Figs. 9, 10 and 11) are both kinematical and dynamical models of the gear systems presented in Fig. 1 (basic version), Figs. 2

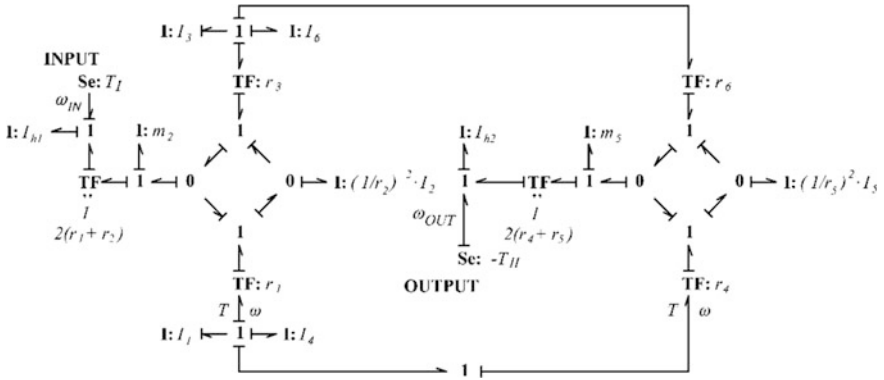


Fig. 9 Bond graph referring to Fig. 1

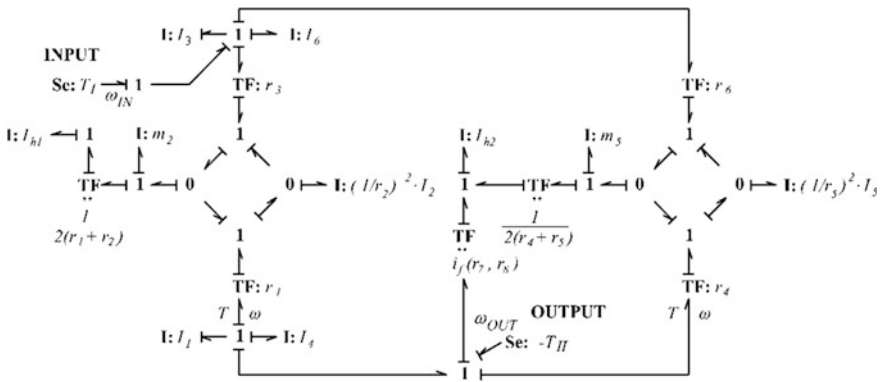


Fig. 10 Bond graph referring to Fig. 2 (input 3-6)

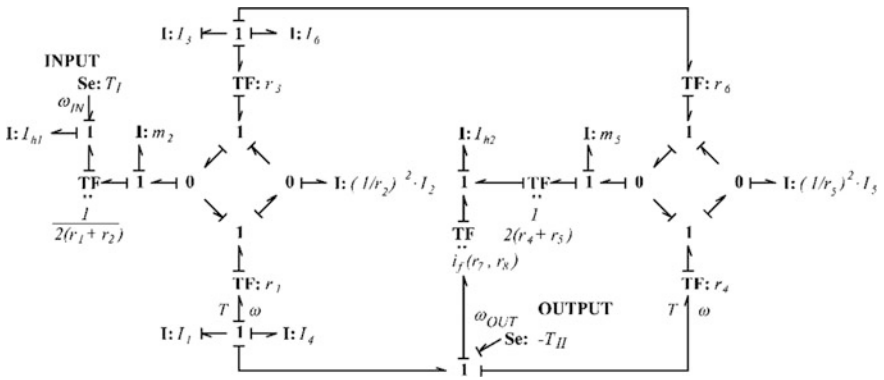


Fig. 11 Bond graph referring to Fig. 3 (input h_1)

and 3 (systems equipped in a friction gear) and Fig. 5 (simplified system). They depict the power flow in the compound system of a planetary gear and a friction gear. However, we can use it for finding out dynamical and kinematical parameters of the system, as well. The dynamical problems will be a subject of future works of the authors of the present paper.

The so-called graph skeletons are built from 1-nodes, 0-nodes and generalized transformer TF. They allow for creation of kinematical model of the gear system. Additional elements I describes inertia of gear parts. The diamond like structures represent I row and II row of gear wheels, respectively.

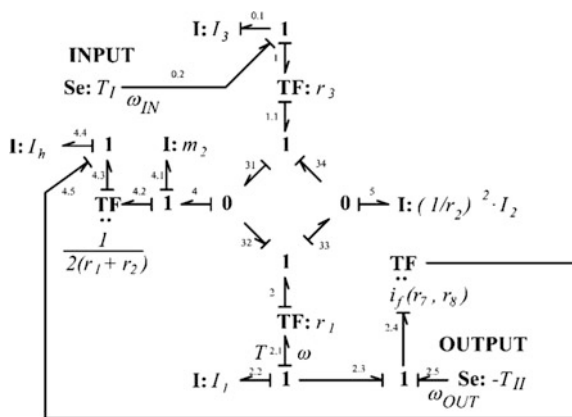
Additional bonds and 1-nodes connect these two structures, because ring gears form the housing of the gear. It is worth noticing that the model presented in Figs. 10 and 11 contains additional bonds describing the friction gear added to the system. The weight of the TF element depicting friction transmission (according to classical graph theory: TF node) depends on radii r_{f1} , r_{f2} and this weight is a variable quantity depending on position of friction parts.

We can derive a set of equations from the bond graph depicted in Fig. 12. Derivation of equations requires numbering of bonds. The bonds should be identifiable via indices, therefore the descriptions of bonds are given in their neighborhoods. The full number of equations consists of 42 items. In formula (26), only the equations needed for obtaining a total ratio i_T are given:

$$\begin{cases} f_4 = r_3 \cdot \omega_{IN} + r_1 \cdot \omega_{OUT} \\ f_{4.5} = \frac{1}{2(r_1 + r_2)} f_4 \\ \omega_{OUT} = \frac{1}{i_T} f_4 \end{cases} \quad (26)$$

The total ratio of the simplified gear set is:

Fig. 12 Bond graph referring to Fig. 5



$$i_T = \frac{\omega_{IN}}{\omega_{OUT}} = \frac{2 \cdot i_f \cdot (r_1 + r_2) - r_1}{r_3}. \quad (27)$$

Substituting $r_i = 0.5 \cdot m \cdot z_i$ (m —gear module), and reducing redundant terms we obtain:

$$i_T = \frac{\omega_{IN}}{\omega_{OUT}} = \frac{2 \cdot i_f \cdot (z_1 + z_2) - z_1}{z_3} = \frac{2 \cdot (-2) \cdot (19 + 50) - 19}{119} = -2.48 \quad (28)$$

Gear module m is the same for all gear wheels in each of the above described design solutions that are discussed in the present paper.

In contradiction to previous graph methods, the considered system having obtained a base on the bond-graphs, is more complicated. However, it allows for versatile further analyses e.g. moments and power flow, whereas in the afore discussed graph models—new systems of equations have to be generated. It is known for contour graphs but for mixed graphs such analyses have not been performed.

4 Conclusions and Final Remarks

The obtained results show usefulness of graph methods for modeling of versatile gear systems—in particular drives of wind turbines. The advantage of graph-based modeling consists in its algorithmic nature and versatility. It allows for avoidance of errors and for comparison of results achieved by several methods. Full compatibility of results obtained by means of different modeling ways has been received which confirms correctness of modeling.

The methods of mixed graphs and contour graphs were modified allowing taking into account friction gear subsystem—new types of edges and arcs were proposed.

The novelty of the considered approach consists in modeling of the system consisting of a compound planetary gear and a friction gear. The mixed multi-graphs as well as modified contour graphs have been utilized which have not yet been proposed.

The Bond graph approach is the final discussed modeling method. Based upon references, it is the widest approach to the discussed problems. The bond graphs applications will be discussed in further papers of the authors, developing co-operation with scientists from Kazakhstan [12, 13].

References

1. Agarwal, S., et al.: Bond graph model of wind turbine blade. In: Bond Graph Modeling. Theory and Practice, Conference 'MathMod', Vienna (2012)

2. Badoud, A.E.: Bond graph algorithms for fault detection and isolation in wind energy conversion. *Arab. J. Sci. Eng.* **39**, 4057–4076 (2014)
3. Bakka, T., Karimi, H.R.: A bond graph approach to modeling and simulation of non-linear wind turbine system. In: *Application of Chaos and Nonlinear Dynamics in Science and Engineering, Understanding Complex Systems*, vol. 3, pp. 41–61 (2013)
4. Bakka, T., Karimi, H.R.: Bond graph modelling and simulation of wind turbine systems. *J. Mech. Sci. Technol.* **27**(6), 1843–1852 (2013)
5. Borutzky, W.: *Bond Graph Methodology: Development and Analysis of Multidisciplinary Dynamic System Models*. Springer, London (2010)
6. Bouscayrol, A., et al.: Energetic macroscopic representation and inversion-based control illustrated on a wind-energy-conversion system using hardware-in-the-loop simulation. *IEEE Trans. Indus. Electron.* **56**(12), 4826–4835 (2009)
7. Drewniak, J., Zawislak, S.: Linear graph and contour graph-based models of planetary gears. *JTAM* **2**, 415–433 (2010)
8. Drewniak, J., Zawislak, S.: Graph-based models of compound planetary gear boxes. *Solid State Phenom.* **199**, 143–148 (2013)
9. Drewniak, J., Zawislak, S., Wieczorek, A.: Modelling multi-way planetary gears by means of contour graphs. *Solid State Phenom.* **220–221**, 126–131 (2015)
10. Geitner, G.-H., Komurgoz, G.: Unified power flow based modeling of power split for Hybrid Electric Vehicles assuming three connections. In: *International Conference on Consumer Electronics, Communications and Networks (CECNet)*, pp. 5464–5469. IEEE (2011)
11. Gomà Ayatsa, J.R.: Hypergraphs for the analysis of complex mechanisms comprising planetary gear trains and other variable or fixed transmissions. *Mech. Mach. Theory* **51**, 217–229 (2012)
12. Ivanov, K.S., Dinassilov A.D., Shingissov B.T.: Self-controlled adaptive gearing of the wind-driven electric plant. The patent for invention No 79226 MU RK, Committee on intellectual property rights of Republic Kazakhstan, Astana, 17 Oct 2012, 13 p. (2012)
13. Ivanov, K.S., Knol, O., Balbaev, G., Shingissov, B.T.: Adaptive transmissions of wind turbine. *Am. J. Mech. Appl.* **2**(6–1), 35–38 (2014)
14. Junco, S., Donaire, A.: Bond graph modeling and simulation of electrical machines. In: *Borutzky, W. (ed.) Bond Graph Modelling of Engineering Systems*, pp. 269–321. Springer Science-Business Media (2011)
15. Karnopp, D.C., Margolis, D.L., Rosenberg, R.C.: *System Dynamics Modeling Simulation and Control of Mechatronic Systems*, 5th edn. Wiley (2012)
16. Samantaray, A.K., et al.: Diagnostic bond graphs for online fault detection and isolation. *Simul. Model. Pract. Theory* **14**(3), 237–262 (2006)
17. Sanchez, R., Medina, A.: Wind turbine model simulation: a bond graph approach. *Simul. Model. Pract. Theory* **41**, 28–45 (2014)
18. Wojnarowski, J., Buchacz, A.: Use of hypergraphs and complete structural numbers in the analysis of vibrating beam systems with non-linearly changing cross-sections. *Vib. Eng.* **3**(4), 593–598 (1989)
19. Wojnarowski, J., Kopeć, J., Zawislak, S.: Gears and graphs. *JTAM* **44**(1), 139–196 (2006)
20. Yutao, L., Di, T.: Dynamics modeling of planetary gear set considering meshing stiffness based on bond graph. *Procedia Eng.* **24**, 850–855 (2011)
21. Zawislak, S.: *The graph-based methodology as an artificial intelligence aid for mechanical engineering design*. Wydawnictwo ATH, Bielsko-Biała (2010)
22. Zhou, W. et al.: Wind-driven generator modeling based on Bond Graph. *Power Syst. Prot. Control* **5** (2010)

Graph-Based Algorithm for the Evaluation of the Mechanical Efficiency of Epicyclic Gear Drive in Hybrid Scooters

P. Coaccioli and E. Pennestri

Abstract Efficiency in transportation of goods and people are crucial objectives for the sustainability of society and the protection of its environment. In the particular case of urban mobility, the current trend is toward the applications of hybrid propulsion, if not purely electric. The growing dimensions of urban areas suggest that a particular attention needs to be addressed to vehicles' energy requirements, air quality and noise pollution. Hybrid technology satisfies the needs of small engines, with low emissions and high energy-efficiency. Therefore, for efficient power management of small internal combustion engines and electric sources, reduced costs and energy efficient gear transmission systems are required. A scheme of hybrid gear transmission, proposed by Sheu, with the aim of minimizing the emissions and fuel consumption, will be herein analysed to estimate the overall mechanical efficiency. In particular, we will discuss an application of a methodology based on graph theory for a systematic and unified analysis of the gear train mechanical efficiency, in all six driving modes. The phases of the proposed analysis are summarized in Table 1.

Keywords Gear train graphs • Hybrid transmission • Planetary gear • Mechanical efficiency • Scooter

P. Coaccioli (✉)

University of Rome Tor Vergata, Via del Politecnico, Rome, Italy
e-mail: p.coaccioli@gmail.com

E. Pennestri

Dipartimento di Ingegneria dell'Impresa, University of Rome Tor Vergata,
Via del Politecnico, Rome, Italy
e-mail: pennestri@mec.uniroma2.it

Table 1 Phases of the proposed mechanical efficiency analysis procedure

I	<ul style="list-style-type: none"> • Complete a kinematic analysis of the PGHT
II	<ul style="list-style-type: none"> • Complete the power flow analysis assuming non-power losses.
III	<ul style="list-style-type: none"> • Taking into account the power flow directions, for each BEGT calculate the efficiency overall η_E.
IV	<ul style="list-style-type: none"> • Taking into account the power flow directions, for each fundamental circuit write eq.(2) the power balance equation.
V	<ul style="list-style-type: none"> • Locate the nodes of input powers and output powers, then apply the power flow balance to each node.
VI	<ul style="list-style-type: none"> • Assign the value of prescribed power. The numbers of prescribed input or output must be equal to the kinematic degree of freedom.
VII	<ul style="list-style-type: none"> • Solve the system of equations obtained
VIII	<ul style="list-style-type: none"> • Verify that the direction of powers in each block is not altered by the presence of friction.
IX	<ul style="list-style-type: none"> • Mechanical efficiency is obtained as the ratio between the sum of output and the sum of input powers.

1 Graph Representation of the Gear Train

The Planetary Gear Hybrid Transmission (PGHT) depicted in Fig. 1 is one of the possible alternatives discussed by Sheu [1, 2]. The engine set includes a 50 cc gasoline motor and the electric unit that functions both as a driver and generator to recharge the batteries. The gear transmission manages these power sources and conveys the motion to the wheels.

The transmission is composed of two epicyclic gear trains with one and two degrees-of-freedom, two clutches (Ci-idling clutch and Ct-shifting clutch) and a brake (B-brake). These elements offer 36 possible configuration modes although very few of them are used.

The presence of a two Planetary Gear Train (PGT), with one and two degrees-of-freedom allows a certain degree of flexibility for the distribution modalities and power system recovery.

Through processing of sensor feedbacks and driver input, the control unit sets functioning modalities that are more suitable to the situation.

As summarized in Table 2, each of the five modes is selected by combining the activation of brake and the two clutches as follows:

- *Electric motor mode*: the power to the wheels is exclusively delivered by the electric motor;
- *Engine mode*: the power to the wheels is exclusively delivered by a fuel engine;

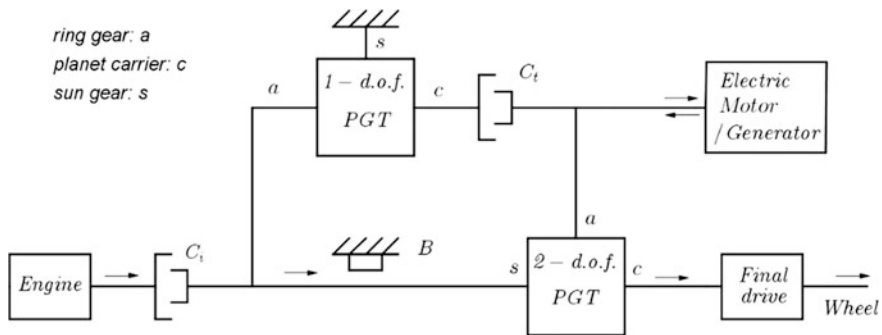


Fig. 1 Configuration of the PGHT

Table 2 Clutches and brake combination for each driving mode

	Idling clutch C_i	Shifting clutch C_t	Brake
Electric mode			X
Engine mode	X	X	
Engine/charging mode	X	X	
Power mode	X		
Regenerative braking mode			X

- *Engine charging mode*: part of the energy is used to move the vehicle, part of it is used to drive the electric motor shaft configured as Generator, with the aim of recharging the batteries;
- *Power Mode*: the power to the wheels is delivered by both engines, simultaneously, through the epicyclic gear with two degrees-of-freedom;
- *Regenerative braking mode*: during the vehicle braking and slowdown, the energy that becomes available through the wheels rotation is conveyed into the electric unit configured as Generator with the aim of recharging the batteries.

2 Kinematic and Power Flow Analysis

The concept herein adopted for analysis is that to each circuit in a mechanism there corresponds a Basic Epicyclic Gear Train (*BEGT*).

Therefore, with the help of gear train-graph correspondence introduced by Freudenstein [3, 4], the topology of this powertrain, depicted in Fig. 2, is initially decomposed into *BEGTs* whose kinematics is governed by the so-called Willis' equation [3–6].

For the *PGHT* under analysis the gears pairs and gear carriers of the *BEGTs*, with their gear ratio R , are summarized in Table 3. In such a table the transfer vertex, denoted by k , corresponds to the gear carrier and the gear wheels are denoted by i and j .

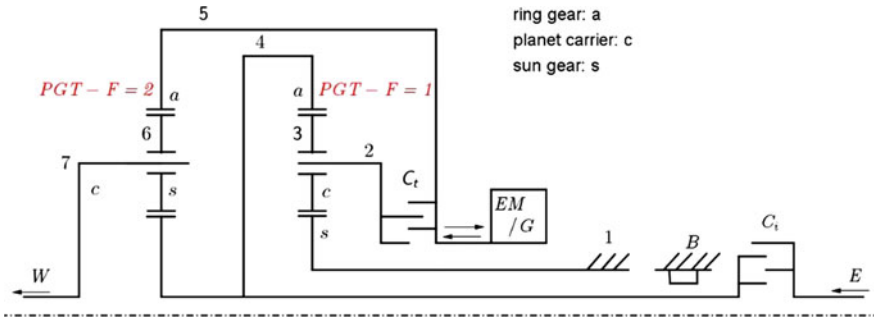


Fig. 2 Functional representation of the *PGHT*

Table 3 Basic epicyclic gear trains (L_i) in the *PGHT*

L_i	i	j	k	R
#1	3	4	2	15/24
#2	3	1	2	24/54
#3	5	6	7	14/62
#4	4	6	7	14/34

The gear train-graphs correspondence is adopted to generate in a systematic way the set of equations whose solution provides the unknown angular speeds [3–6].

Since the gear train topology depends on activation of the clutches and brakes, the corresponding graphs are depicted for each driving mode. Vertices represent links and edges kinematic pairs, respectively. Dotted lines denote gear pairs whereas continuous edges revolute pairs. The Electric and the Regenerative mode have the same graph representation (see Fig. 3a), as well as the Engine and the Engine/Charging mode (see Fig. 3b). In Fig. 3c is shown the Power mode graph, where only the idling clutch is selected.

On the basis of the methodology of Pennestri and Freudenstein [6], power flow distribution under different power transmission modes is computed. It is assumed that the working conditions are stationary with inertia and friction effects neglected.

For each driving mode a mechanism graph can be drawn. The fundamental circuits in the graph are identified and each represented by means of a block. The nodes connecting two blocks represent a link shared by the circuits. The power leaving the blocks has a negative algebraic sign whereas it is positive for the power entering the blocks. The convention for interpreting the powers algebraic sign is switched for nodes.

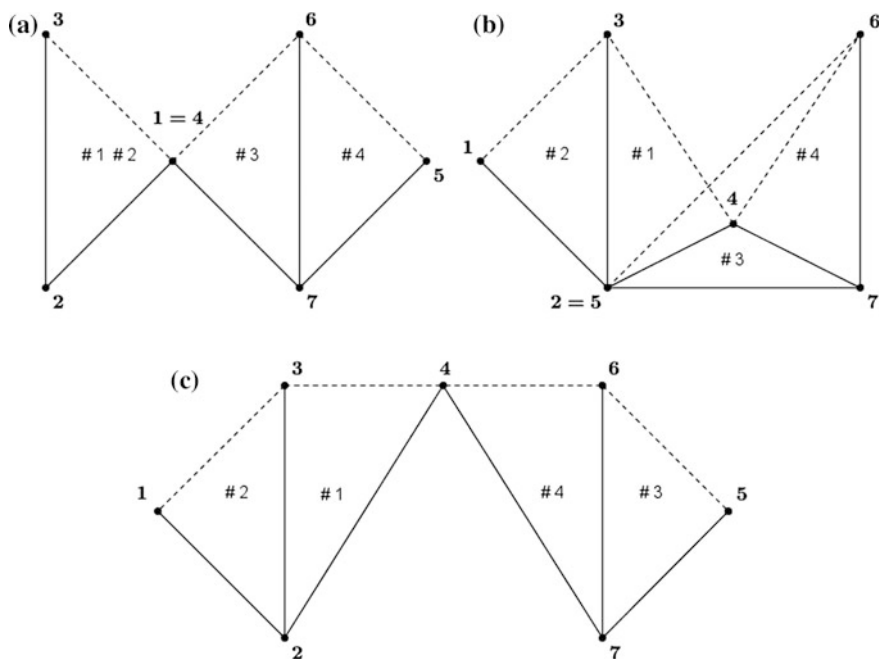


Fig. 3 Graph of the gear train in electric/mode and regenerative braking/mode (a), in engine and engine-charging mode (b) and power mode (c)

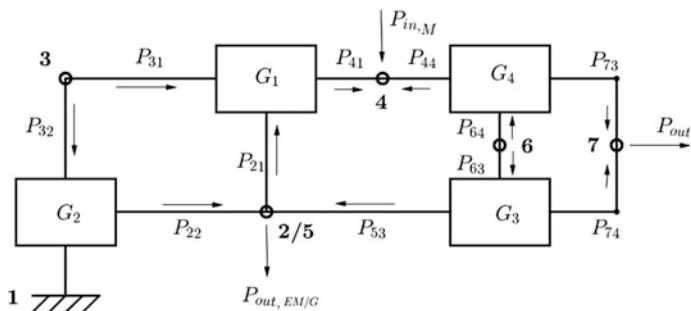


Fig. 4 Block schematization for the engine/charging mode

Table 4 Results of kinematic analysis of the hybrid transmission represented in Fig. 2 in eng/charg mode

ω_1 (rpm)	ω_2 (rpm)	ω_3 (rpm)	ω_4 (rpm)	ω_5 (rpm)	ω_6 (rpm)	ω_7 (rpm)
/	5005	7270	8500	5005	-10,227	3055

Table 5 Results of power flow analysis of the hybrid transmission represented in Fig. 2 in eng/charg mode

$P_{21} = -699.9$ W	$P_{22} = -83.8$ W	
$P_{31} = -245.9$ W	$P_{32} = -245.9$ W	
$P_{41} = 781.1$ W	$P_{44} = 1140.0$ W	$P_{53} = 798.2$ W
$P_{63} = -919.0$ W	$P_{64} = -919.0$ W	
$P_{73} = -1380.3$ W	$P_{74} = -724.8$ W	$P_{out} = 1850$ W

In Fig. 4 the power flow diagram of the gear train in Engine/Charging Mode is represented. Also Table 4 shows the kinematic characteristics of the PGHT's members in that modality, where the input was the angular velocity of the member #4. The results of the power flow analysis are summarized in Table 5.

3 PGHT Mechanical Efficiency

The evaluation of *PGHT* mechanical efficiency requires consideration of the various power distribution modes and the efficiency of the *BEGTs*. For this purpose, *BEGTs* with one and two degrees-of-freedom need to be distinguished.

In the first case, to calculate the efficiency of spur gear trains with a fixed gear carrier and the flowing power from i to j , the following formula [7] is adopted:

$$\eta_{ij}^0 = 1 - \left| \frac{1}{5} \left(\frac{1}{z_i} \pm \frac{1}{z_j} \right) \right| \tag{1}$$

where: z_i and z_j denote the gear teeth for wheels i and j , respectively. The (+) algebraic sign holds for the external wheels and (-) for the internal wheels.

Table 6 summarizes the mechanical efficiencies for the different kinematic inversions of a simple ordinary gear train.

Regarding the efficiency calculation of the *BEGT* with two degrees-of-freedom, by means of the method of Pennestri et al. [5, 8, 9], the mechanical efficiency is computed as a linear combination of angular speed ratios. Then it is necessary to assess if the Basic Epicyclic Gear Train (*BEGT*) works as *DISO* (dual input-single output) or *SIDO* (single input-dual output). Table 7 shows all the expressions for

Table 6 Efficiencies of single epicyclic spur-gear trains

Transmission ratio condition	Driving	Driven	Fixed	η
	i	j	k	η_{ij}^0
	j	i	k	η_{ij}^0
$(R < 0)(R > 1)$	i	k	j	$\frac{R\eta_{ij}^0 - 1}{R - 1}$
$(0 < R < 1)$	i	k	j	$\frac{R - \eta_{ij}^0}{\eta_{ij}^0(R - 1)}$
$(R < 0)(R > 1)$	k	i	j	$\frac{(R - 1)\eta_{ij}^0}{R - \eta_{ij}^0}$
$(0 < R < 1)$	k	i	j	$\frac{R - 1}{\eta_{ij}^0 R - 1}$
$(R < 1)$	j	k	i	$\frac{R - \eta_{ij}^0}{R - 1}$
$(R > 1)$	j	k	i	$\frac{1 - R\eta_{ij}^0}{\eta_{ij}^0(1 - R)}$
$(R < 1)$	k	j	i	$\frac{(R - 1)\eta_{ij}^0}{R\eta_{ij}^0 - 1}$
$(R > 1)$	k	j	i	$\frac{R - 1}{R - \eta_{ij}^0}$

Table 7 Mechanical efficiencies of 2 degree-of-freedom basic epicyclic gear train

Case	Driving links	Driven links	η
DISO	i, k	j	$\frac{\omega_j}{\frac{(R-1)\omega_k}{R\eta_{i(k-j)}} + \frac{1}{R\eta_{k(i-j)}}\omega_i}$
	j, k	i	$\frac{\omega_i}{\frac{R}{\eta_{k(j-i)}}\omega_j + \frac{(R-1)}{\eta_{j(k-i)}}\omega_k}$
	i, j	k	$\frac{\omega_k}{\frac{1}{(1-R)\eta_{j(i-k)}}\omega_i + \frac{R}{(R-1)\eta_{i(j-k)}}\omega_j}$
SIDO	i	j, k	$\frac{\eta_{j(i-k)}(1-R)\omega_k + \eta_{k(i-j)}R\omega_j}{\omega_i}$
	j	i, k	$\frac{\eta_{k(j-i)}\omega_i + \eta_{i(j-k)}(R-1)\omega_k}{R\omega_j}$
	k	i, j	$\frac{\eta_{i(k-j)}R\omega_j - \eta_{j(k-i)}\omega_i}{(R-1)\omega_k}$

Table 8 Overall efficiency of the *PGHT*, for each *BEGT* and modality

	η				
<i>BEGT</i>	Electric	Engine	Eng/charg	Power	Reg/brak
#1	\	0.962	0.962	0.962	\
#2	\	0.998	0.998	0.998	\
#3	0.963	0.960	0.960	0.903	0.984
#4	\	0.969	0.969	0.996	\

Table 9 Mechanical efficiency of the planetary gear hybrid transmission

Modality of power transmission	η
Electric	0.916
Engine	0.914
Engine/charging	0.909
Power	0.895
Regenerative/braking	0.834

the computation of such factors. By the notation $\eta_{a(b-c)}$ is meant the mechanical efficiency of the *BEGT* with link ‘*a*’ fixed and power flow directed from ‘*b*’ to ‘*c*’. For each driving mode, the estimated *BEGT* mechanical efficiencies are summarized in Table 8.

We need to consider network losses, or meshing losses, in order to assess the *PGHT* performance in terms of mechanical efficiency. The network losses are expressed through coefficients α and β inserted within a simple power output:

$$\alpha P_i + \beta P_j = 0. \quad (2)$$

Coefficients α and β are parameters calculated as a function of each fundamental circuit or *BEGT* parameters and kinematics and taking into account power flow, earlier computed within the Power Flow Analysis phase. Similarly with the previous analysis, the power balance is imposed at nodes and for each circuit, with the exception of the *BEGT*, where a member is fixed. In such a condition only the power balance equation with meshing losses is considered.

After power balance equations at nodes are deduced, the system of analysis equations is complete and can be solved.

The mechanical efficiency (numerical results summarized in Table 9) for the hybrid gear transmission under analysis is calculated, assuming positive values for powers, with the following equation:

$$\eta = \frac{P_{output}}{P_{input}}. \quad (3)$$

4 Conclusions

The main novelty of this paper is the application of a graph-based methodology to the mechanical efficiency analysis of the hybrid scooters planetary gear trains. To the best of the authors' knowledge, the analysis herein presented was not available for light-duty vehicles such as hybrid scooters. Graph theory allows a systematic setup of analysis equations and the possibility of topology optimization. The analysis assesses the transmission energy performances of a gear train installed in a hybrid scooter. The obtained results, considered also simplicity and vehicle compactness, justify and confirm the choice of addressing the urban transportation towards the hybrid solution also for small vehicles. It is essential that we have the easiness of energy recovery in the braking process with the aim of empowering the autonomy and/or, at the same time, allowing us to contain the dimension (i.e. weight and costs) of the batteries. In an urban environment, conditions of activating the pure electrical mode are several. One can observe that the operating mode with the worst energy condition is Regenerative/Braking. However, one should consider that in this case most of the kinetic energy from the wheels, after a conversion into electric energy, is stored in batteries. However, this operating mode is only a small percentage of the vehicle's life. More important are the Electric and Engine modes. In the Electric mode, the vehicle autonomy heavily depends on battery capacity and low power losses of all mechanical components. Therefore, in these modes the highest mechanical efficiency must be required at design stage.

References

1. Sheu, K.-B.: Conceptual design of hybrid scooter transmission with planetary gear-trains. *Appl. Energy* **84**, 526–541 (2007)
2. Sheu, K.-B.: Simulation for the analysis of an hybrid electric scooter powertrain. *Appl. Energy* **85**, 590–606 (2008)
3. Freudenstein, F.: An application of boolean algebra to the motion of epicyclic drives. *ASME J. Eng. Ind.* **93**(Series-B), 176–182 (1971)
4. Freudenstein, F., Yang, A.T.: Kinematics and statics of a coupled spur-gear train. *Mech. Mach. Theory* **7**, 263–275 (1972)
5. Pennestri, E., Mariti, L., Valentini, P.P., Mucino, V.H.: Efficiency evaluation of gearboxes for parallel hybrid vehicles. *Mech. Mach. Theory* **49**, 158–176 (2012)
6. Pennestri, E., Freudenstein, F.: A systematic approach to power-flow and static-force analysis in epicyclic spur-gear trains. *ASME J. Mech. Des.* **115**, 639–644 (1993)
7. Tuplin, W.: *Involute gear geometry*. Chatto Windus, London (1962)
8. Pennestri, E., Freudenstein, F.: The mechanical efficiency of epicyclic gear trains. *ASME J. Mech. Des.* **115**, 645–651 (1993)
9. Pennestri, E., Valentini, P.P.: A review of formulas for the mechanical efficiency analysis of two-degrees-of-freedom epicyclic gear trains. *ASME J. Mech. Des.* **125**(3), 602–608 (2003)

Graph-Theoretic Modelling and Sensitivity Analysis of Dynamic Systems

J. Banerjee and J. McPhee

Abstract In this paper, a graph-theoretic formulation is presented that can be used to generate governing equations for dynamic systems in a flexible, automated and efficient fashion. Furthermore, a modified formulation is presented to demonstrate the applicability of graph-theoretic methods to generate the sensitivity equations for dynamic systems. An example is provided, in which the new graph-theoretic formulation is used to generate the governing and sensitivity equations simultaneously from the linear graph representation of the systems.

Keywords Linear graph · Sensitivity analysis · Direct differentiation · Symbolic equations

1 Introduction

One of the first uses of a linear graph was done by the famous mathematician Leonhard Euler in 1736 [1]. Later, researchers like Koenig et al. [2] extended the use of a linear-graph as a unified system modelling theory. Subsequently, graph-theoretic formulations have been successfully applied to model electrical circuits [3], mechanical/mechatronics systems [4], hydro-mechanical [5] and electro-chemical systems [6]. In this paper, a graph-theoretic formulation will be presented for automated generation of governing and sensitivity equations. First, basics of graph-theoretic modelling methods and analytical sensitivity studies will be introduced. Next, the process of graph-theoretic generation of governing equations and sensitivity equations will be presented. Finally, the theory presented in this paper will be further clarified using an example problem.

J. Banerjee (✉) · J. McPhee
University of Waterloo, Waterloo, Canada
e-mail: jbanerjee83@gmail.com

1.1 Graph-Theoretic Modelling of Dynamic Systems

To present the process of graph-theoretic modelling and sensitivity analysis, it is necessary to define certain key concepts. Figure 1 shows a simple spring-mass-damper system and the corresponding linear graph representation. The linear graph is comprised of nodes (G and A) and edges (k, c, m, F, and s). For multibody systems, nodes represent frames of reference. In Fig. 1, the node G represents the frame of reference fixed to the ground, whereas the node A refers to the frame of reference fixed to the center of the mass of the body. For other systems, (e.g. electrical circuits, hydraulic circuits), the nodes represent specific points in the system where measurements are made.

The edges of a linear graph represent various measurements, and their arrow directions denote the positive directions for those measurements. Depending on the nature of these quantities they are divided into two broad categories called through variables (e.g. current, flow rate, force, and torque) and the across variables (e.g. voltage difference, pressure difference, position, velocity, acceleration etc.). Functionally, these measurements are often associated with various components that constitute the system. In the linear graph shown above, the edge k corresponds to the spring, the edge m corresponds to inertia, c refers to the damper, F refers to the applied force, and the edge s corresponds to the slider joint.

In multibody systems, the edges are associated with through and across variables from both translational and rotational domains. On the other hand, for scalar systems (e.g. electrical circuits), the edges are usually associated with one through variable (current) and one across variable (voltage difference).

To continue this presentation, a few key terms need to be introduced. In the linear graph G shown in Fig. 1, which has $e = 5$ edges and $n = 2$ nodes, the following terms are defined.

A **tree** of a linear graph G is a connected sub-graph of G that contains all the nodes of G and has only one unique path between any two nodes. Edges of the tree are known as branches. A tree in G can have $w = n - 1$ branches. In Fig. 1, one possible tree selection could be the edge m. A **cotree** is the part of the graph G which remains after removing the tree. The edges of a cotree are called chords. In Fig. 1, if edge m is selected as the tree, the edges k, c, F, and s will become chords. In any graph, there can be $u = e - n + 1$ chords in the cotree.

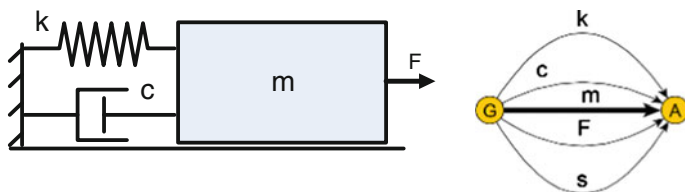


Fig. 1 Spring-mass-damper system and the corresponding linear graph

A **cutset** is a subset of the graph G , which when removed from G , divides it into two parts in such a way that no subset of it can exist. An **f-cutset** is a cutset that contains exactly one branch and a unique set of chords. In any graph G , there can be 'w' f-cutsets.

A **circuit** is a sub-graph of G where every pair of nodes has exactly two distinct paths between them. In Fig. 1, the edges m and c form a circuit. An **f-circuit** is a circuit in G , which contains exactly one chord and a unique set of branches. In any graph G , there can be 'u' f-circuits.

The **f-cutsets** and the **f-circuits** are dependent on the choice of the tree branches. In the tree structure shown in Fig. 1 in bold lines, there is one f-cutset (the edges $[m, k, c, F, s]$) and four f-circuits (the edges $[k, m]$, $[c, m]$, $[F, m]$, $[s, m]$).

Using these properties of a linear graph, a graph-theoretic method can be formulated to capture the topology of any dynamic system and derive both the governing and sensitivity equations for it.

1.2 Sensitivity Analysis of Dynamic Systems

Sensitivity analysis refers to the study of changes in system behavior brought in by the changes in entities inherent to the system. Mathematically, it is a problem of finding the derivatives of the state variables of a system with respect to the system parameters. It is routinely used in many engineering applications like design and optimization of physical systems, model simplification, and optimal control. For complex and/or large scale systems, however, this process can become complicated.

For models where symbolic equations are available, sensitivity analysis can be performed by the direct differentiation method. In this method a set of sensitivity equations are derived from the governing equations by symbolic differentiation. By solving these sensitivity equations, the corresponding sensitivity information is obtained as functions of time.

Direct differentiation sensitivity analysis is portable, easy to implement, stable, and produces results that are numerically exact. This method has been used to perform sensitivity analysis for systems governed by kinematic equations [7], differential equations [8–10] and also differential-algebraic equations [11].

Since it requires the solution of an increased number of equations, this method can become intractable for systems with a large number of parameters, especially if the size of the system is also large. To address this drawback, a graph-theoretic framework has been developed by Banerjee and McPhee [12]. In this formulation, the sensitivity equations are generated directly from the linear graph of the system using the very procedure used to generate the governing equations.

The graph-theoretic formulation uses direct differentiation as the underlying method, but instead of symbolically differentiating the governing equations after the fact, it stores the pre-formed expressions of the derivatives as sensitivity constitutive equations and evaluates them as needed. Figures 2 and 3 summarize the difference between direct differentiation and the proposed graph-theoretic sensitivity

Fig. 2 Direct differentiation method

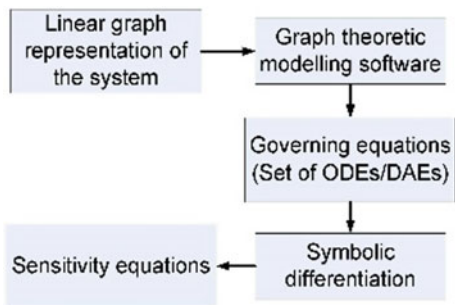
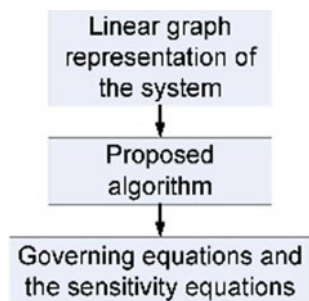


Fig. 3 Graph-theoretic method



analysis. The following sections will illustrate the process of generation of the governing equations and the sensitivity equations.

2 Generation of Governing and Sensitivity Equations

The graph-theoretic formulation presented by Banerjee and McPhee [12] uses two topologically identical linear graphs G_1 and G_2 (vide Fig. 1) to generate the required equations. The through variables associated with the edges of the graph G_1 are the forces/torques acting between the two frames of reference (G and A) whereas the associated across variables are the position, velocity, and acceleration (both translational and rotational) of frame A with respect to frame G. The through and across variables associated with the edges of graph G_2 (sensitivity graph) are defined to be the derivatives of the variables associated with G_1 with respect to the desired model parameter. A complete list of translational through and across variables are given in Table 1, where the subscripts used in the variable names denote the differentiation with respect to the model parameter appearing as a subscript.

In any dynamic system, two types of equations govern these through and across variables. These can be either topological equations, which are equations that describe how through variables (or across variables) of different edges are related to

Table 1 Translational through and across variables for G_1 and G_2

	Through variables	Across variables
G_1	$\boldsymbol{\tau} = \{ \bar{F}^m \quad \bar{F}^k \quad \bar{F}^c \quad \bar{F}^F \quad \bar{F}^s \}^T$	$\boldsymbol{\alpha} = \{ \bar{r}^m \quad \bar{r}^k \quad \bar{r}^c \quad \bar{r}^F \quad \bar{r}^s \}^T$
G_2	$\boldsymbol{\tau}_b = \{ \bar{F}_k^m \quad \bar{F}_k^k \quad \bar{F}_k^c \quad \bar{F}_k^F \quad \bar{F}_k^s \}^T$	$\boldsymbol{\alpha}_b = \{ \bar{r}_k^m \quad \bar{r}_k^k \quad \bar{r}_k^c \quad \bar{r}_k^F \quad \bar{r}_k^s \}^T$

each other, or constitutive equations, which are equations that describe how the through variables are related to the across variables. Functionally, the topological equations describe the connectivity of the different components of the system and can be readily derived from the linear graph of the system, whereas the constitutive equations describe the physical nature of the components and must be specified prior to the derivation.

To derive a compact form of equations, this paper will use a branch-chord formulation, where the first step is the selection of two trees for the graphs G_1 and G_2 . For this demonstration, the edge m is selected as the tree for both trees and for both translational and rotational domains (bold line). This formulation uses a particular form of topological equations known as the f-cutset equations (for through variables) and f-circuit equations (for across variables). For the graph shown in Fig. 1, the translational f-cutset equations for the graphs G_1 and G_2 are shown in Eq. (1).

$$\begin{aligned}
 G_1: \bar{F}^m + \bar{F}^k + \bar{F}^c + \bar{F}^F + \bar{F}^s &= 0, \\
 G_2: \bar{F}_k^m + \bar{F}_k^k + \bar{F}_k^c + \bar{F}_k^F + \bar{F}_k^s &= 0.
 \end{aligned}
 \tag{1}$$

These equations describe how the different through variables are related to each other, as constrained by the topology of the system. Similar equations governing the across variables are known as f-circuit equations. The translational f-circuit equations are shown in Eq. (2).

$$\begin{aligned}
 G_1: \{ -\bar{r}^m + \bar{r}^i = 0 \} \{ -\dot{\bar{r}}^m + \dot{\bar{r}}^i = 0 \} \{ -\ddot{\bar{r}}^m + \ddot{\bar{r}}^i = 0 \} \quad i = k, c, F, s, \\
 G_2: \{ -\bar{r}_k^m + \bar{r}_k^i = 0 \} \{ -\dot{\bar{r}}_k^m + \dot{\bar{r}}_k^i = 0 \} \{ -\ddot{\bar{r}}_k^m + \ddot{\bar{r}}_k^i = 0 \} \quad i = k, c, F, s.
 \end{aligned}
 \tag{2}$$

The main benefit of using a branch-chord formulation is that it makes it possible to express the branch through variables in terms of the chord through variables (using f-cutset equations) and to express the chord across variables in terms of the branch across variables (using f-circuit equations). This opens up the possibility of reducing, by substitution, the number of unknowns in the generated equations. Further details are provided by Banerjee [13].

As mentioned before, the constitutive equations for the system must be specified prior to the derivation process. These equations govern how the through variables are related to the across variables depending on the nature of the associated components. For multibody systems, it is necessary to define these equations for both translational and rotational domains. A list of the translational constitutive equations for the graph G_1 is given in Eq. (3).

$$\begin{aligned}\bar{F}^m &= -m\ddot{r}^m & \bar{F}^F &= F(t)\hat{i} & \bar{r}^m &= x(t)\hat{i}, \\ \bar{F}^c &= -c\dot{r}^c & \bar{F}^k &= -k(\bar{r}^k - l_0\hat{i}) & \bar{F}^s &= 0.\end{aligned}\quad (3)$$

For graph-theoretic sensitivity analysis the equations associated with G_2 is defined as the derivatives of those from G_1 with respect to the model parameter. Assuming the spring constant k as the model parameter the constitutive equations can be written down as shown in Eq. (4), (subscripts refer to derivatives).

$$\begin{aligned}\bar{F}_k^m &= -m\ddot{r}_k^m & \bar{F}_k^F &= 0 & \bar{r}_k^m &= x_k(t)\hat{i}, \\ \bar{F}_k^c &= -c\dot{r}_k^c & \bar{F}_k^k &= -(\bar{r}^k - l_0\hat{i}) - k\bar{r}_k^k & \bar{F}_k^s &= 0.\end{aligned}\quad (4)$$

Apart from the constitutive equations, to capture the nature of the constraint the joints enforce on the system, it is also necessary to specify the motion spaces and the reaction spaces for the joints. The motion space of a joint is defined as the span of directions along which motions are allowed by that joint and the reaction spaces are the span of directions along which the joint does not allow motion to take place and offer reactive forces and torques. For example, in an ideal prismatic joint, the translational motion space is the sliding direction, and the rotational motion space is null. At the same time, its translational reaction space contains the two directions normal to the slider direction and the rotational reaction space contains all three of the principal directions.

To generate the equations using graph-theoretic formulation, the first step is the substitution of the constitutive equations into the f-cutset equations obtained from the rigid bodies or joints present in the tree and the projection (scalar product) of the resulting equations on to the corresponding motion spaces. The projected equations are shown in Eq. (5). Since the rotational motion space is null, the rotational f-cutset equations do not generate any new equation in this step.

$$\begin{aligned}G_1 &: (-m\ddot{r}^m - k(\bar{r}^k - l_0\hat{i}) - c\dot{r}^c + F(t)\hat{i} = 0) \cdot \hat{i}, \\ G_2 &: (-m\ddot{r}_k^m - (\bar{r}^k - l_0\hat{i}) - k\bar{r}_k^k - c\dot{r}_k^c = 0) \cdot \hat{i}.\end{aligned}\quad (5)$$

The next step is the elimination of the secondary variables (i.e. branch through variables and chord across variables). Using the equations shown in (1)–(2), the chord across variables (i.e. across variables from the edges k , c , F , and s) are replaced with the branch across variables (i.e. across variables from the edge m). This results in the equations shown in Eq. (6).

$$\begin{aligned}G_1 &: (-m\ddot{r}^m - k(\bar{r}^m - l_0\hat{i}) - c\dot{r}^m + F(t)\hat{i} = 0) \cdot \hat{i}, \\ G_2 &: (-m\ddot{r}_k^m - (\bar{r}^m - l_0\hat{i}) - k\bar{r}_k^m - c\dot{r}_k^m = 0) \cdot \hat{i}.\end{aligned}\quad (6)$$

Using the constitutive equations one more time, Eq. (6) can be readily simplified into two second-order ordinary differential equations as shown in equation

$$\begin{aligned}
 G_1 : -m\ddot{x} - k(x - l_0) - c\dot{x} + F(t) &= 0 \Rightarrow m\ddot{x} + k(x - l_0) + c\dot{x} = F(t), \\
 G_2 : -m\ddot{x}_k - (x - l_0) - kx_k - c\dot{x}_k &= 0 \Rightarrow m\ddot{x}_k + (x - l_0) + kx_k + c\dot{x}_k = 0.
 \end{aligned}
 \tag{7}$$

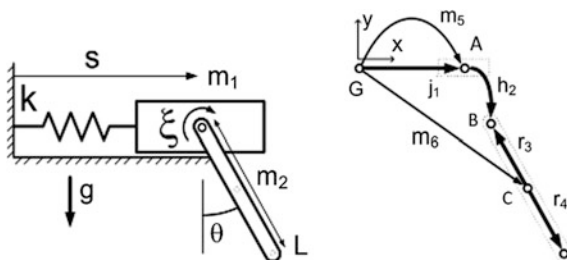
The second equation in (7) is the sensitivity equation of the system with respect to the model parameter k . This demonstrates that the presented graph-theoretic formulation can generate the governing equations and the sensitivity equations directly from the linear graph simultaneously. If sensitivities with respect to a second parameter is required, a third graph G_3 can be defined to generate the corresponding equations. Further details of this process can be found in reference [12].

3 Example Problem

Figure 4 shows a simple mechanism with its linear graph representation. It constitutes two rigid bodies of masses $m_1 = 1$ kg and $m_2 = 5$ kg, connected by a revolute joint that includes a torsional damper with damping coefficient $\xi = 10$ N m s/rad. The parameters $L = 1$ m and $r = 0.03$ m represent the length and radius of cross-section of the pendulum. The sliding block is allowed to move along a prismatic joint and is connected to a linear spring of spring constant $k = 50$ N/m and unstretched length $s_0 = 0.50$ m. The gravity $g = 9.81$ m/s² acts in the downward direction. The nodes A and C represent the frames of reference fixed to the two centers of mass, G refers to the ground, and B refers to the frame of reference fixed to the body m_2 at the hinge point of the pendulum. The edges m_5, m_6 correspond to the inertia, j_1 refers to the slider joint, h_2 is the revolute joint, r_3 and r_4 are body fixed vectors.

In branch-chord formulation, the choice of tree determines the state variables of the system. For the system shown in Fig. 4, if the edges $m_5, m_6, r_3,$ and r_4 are selected as branches (for both translational and rotational domain), the equations are generated in terms of the absolute coordinates (i.e. position and orientation of the two bodies), but if the edges j_1, h_2, r_3 and r_4 are selected as branches (both translational and rotational), the equations are generated in terms of the slider displacement s and the revolute joint angle.

Fig. 4 Slider-pendulum with linear graph



For this example the latter choice is used for the equation generation. By using the formulation presented in this paper, the governing equations are generated for this system as shown in Eq. (8).

$$\begin{aligned} m_2(r^2/4 + L^2/3)\ddot{\theta} + m_2(L/2) \cos \theta \ddot{s} + m_2g(L/2) \sin \theta + \zeta \dot{\theta} &= 0, \\ m_2(L/2) \cos \theta \ddot{\theta} + (m_1 + m_2)\ddot{s} - m_2(L/2) \sin \theta \dot{\theta}^2 + k(s - s_0) &= 0. \end{aligned} \quad (8)$$

By selecting the parameter L for sensitivity analysis the sensitivity graph G_2 results in the equations shown in Eq. (9).

$$\begin{aligned} m_2(r^2/4 + L^2/3)\ddot{\theta}_L + \frac{m_2}{2}(\cos \theta - L \sin \theta \theta_L)\ddot{s} + m_2(L/2) \cos \theta \ddot{s}_L + \Upsilon_3 &= 0 \\ m_1 \ddot{s}_L + m_2(\ddot{s}_L + (L/2) \cos \theta \ddot{\theta}_L + \Upsilon_1) + ks_L &= 0 \end{aligned} \quad (9)$$

$$\begin{aligned} \Upsilon_3 &= 2m_2(L/3)\ddot{\theta} + (m_2/2)g \sin \theta + m_2g(L/2) \cos \theta \theta_L + \zeta \dot{\theta}_L \\ \Upsilon_1 &= (1/2)(\cos \theta - L \sin \theta \theta_L)\ddot{\theta} - (L \sin \theta \dot{\theta})\dot{\theta}_L - (1/2)(\sin \theta + L \cos \theta \theta_L)\dot{\theta}^2. \end{aligned} \quad (10)$$

It can be readily verified that the equations shown in (9) are the symbolic derivatives of the governing equations shown in Eq. (8). The initial conditions for the combined Eqs. (8) and (9) are shown in Eq. (11). The rest of the state variables are initiated at zero. The Runge-Kutta-Fehlberg method was used for numerical simulation.

$$s(0) = 0.5 \text{ m} \quad \dot{s}(0) = 1 \text{ m/s}. \quad (11)$$

Figures 5 and 6 clearly identify the response of the system and validate the solution of the generated sensitivity equations. Figure 5 shows that the values of $s(t)$ and $\theta(t)$ settles after initial oscillations, which is expected for the damped system. The behavior of $s_L(t)$ can also be explained by considering the structure of the system. By considering the configuration of the system, it can be clearly seen that for different values of the parameter L, the resulting output of $s(t)$ always

Fig. 5 Simulation results

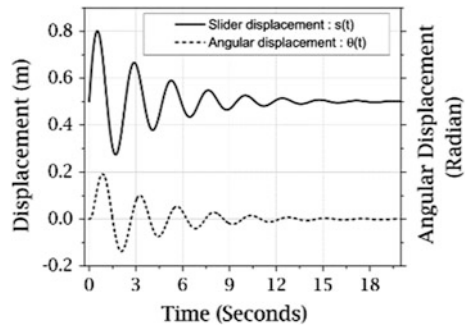
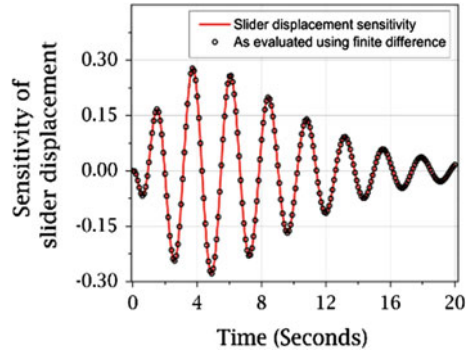


Fig. 6 Sensitivity s_L versus time



approaches the same datum level s_0 asymptotically. As a result the sensitivity of $s(t)$ with respect to the parameter L also oscillates and then settles to zero. To validate the sensitivity results the solution of the sensitivity equations were compared with those obtained by finite difference formulation. The comparison, as shown in Fig. 6, validates the sensitivity results obtained through the presented graph-theoretic formulation.

4 Conclusions

A graph-theoretic formulation is presented in this paper that can be used to generate governing equations for dynamic systems in an automated and algorithmic fashion. The graph-theoretic method has been demonstrated to be able to derive equations from a pictorial description of the system called the linear graph. The method has been demonstrated in details using the example of a spring-mass-damper system. Concurrently, a graph-theoretic framework is presented that can be used to perform symbolic sensitivity analysis of dynamic systems. This framework is capable of generating symbolic sensitivity equations for the systems using the same algorithmic steps used in traditional graph-theoretic modelling techniques. The simultaneous generation of governing and sensitivity equations have been presented using an example of a sliding pendulum.

Acknowledgments Toyota, Maplesoft, and NSERC Canada.

References

1. Biggs, N., Lloyd, E.K., Wilson, R.J.: Graph Theory 1736–1936. Clarendon Press, New York, NY, USA (1986)
2. Koenig, H.E., Tokad, Y., Kesavan, H.K.: Analysis of Discrete Physical Systems. McGraw-Hill, New York (1967)

3. Seshu, S., Reed, M.B.: *Linear Graphs and Electrical Networks*. Addison-Wesley Pub. Co., Boston (1961)
4. McPhee, J.J.: On the use of linear graph theory in multibody system dynamics. *Nonlinear Dyn.* **9**, 73–90 (1996)
5. Gupta, R., Prasad, T.D.: Extended use of linear graph theory for analysis of pipe networks. *J. Hydraulic Eng.* **126**(1), 56–62 (2000)
6. Dao, T.S., McPhee, J.J.: Dynamic modeling of electrochemical systems using linear graph theory. *J. Power Sources* **196**(23), 10442–10454 (2011)
7. Pesch, V.J., Hinkle, C.L., Tortorelli, D.A.: Optimization of planar mechanism kinematics with symbolic computation. In: *IUTAM Symposium on Optimization of Mechanical Systems*, Stuttgart, Germany, pp. 221–230 (1996)
8. Serban, R., Freeman, J.: Identification and identifiability of unknown parameters in multi-body dynamic systems. *Multibody Syst. Dyn.* **5**, 335–350 (2001)
9. Carr, S.M., Savage, G.J.: Symbolic sensitivity analysis of nonlinear physical systems using graph-theoretical modeling (1995)
10. Savage, G.J.: Automatic formulation of higher order sensitivity models. *Civil Eng. Syst.* **10**(4), 335–350 (1993)
11. Serban, R., Freeman, J.S.: Direct differentiation methods for the design sensitivity of multi-body dynamic systems. In: *The 1996 ASME Design Engineering Technical Conferences and Computers in Engineering Conference*, pp. 18–22 (1996)
12. Banerjee, J.M., McPhee, J.J.: Graph-theoretic sensitivity analysis of multibody systems. *J. Comput. Nonlinear Dyn.* **9**(4) (2014)
13. Banerjee, J.: Graph-theoretic sensitivity analysis of dynamic systems. Ph.D. thesis, University of Waterloo, Canada (2013)

Three-Dimensional Analysis of Vehicle Stability Using Graph Theory

G.G. Moreno, R.L.P. Barreto, R.S. Vieira, L. Nicolazzi
and D. Martins

Abstract Vehicle stability is a widely studied topic today. It is crucial that we develop a better understanding of one of the main problems of vehicular accident rate problems throughout the world: the rollover accident. The main goal for researchers is to determine a way to predict vehicle behaviour under a variety of circumstances. Davies method is a mathematical tool that allows the static and kinematic analysis of any kind of mechanisms, as well as we can find in vehicle suspensions. This method uses the Graph theory that enables kinematic chain representation by means of a graph for later analysis. In this paper we present the vehicle stability kinematic analysis using Graph theory, Screw theory and the Davies method.

Keywords Static stability factor · Graph theory · Davies method · Road safety · Vehicle rollover

G.G. Moreno (✉) · R.L.P. Barreto · R.S. Vieira · L. Nicolazzi · D. Martins
Department of Mechanical Engineering, Federal University of Santa Catarina,
88040-900 Florianópolis, Brazil
e-mail: gmoren@hotmail.com

R.S. Vieira
e-mail: rvieira@grante.ufsc.br

L. Nicolazzi
e-mail: lauro@grante.ufsc.br

D. Martins
e-mail: daniel.martins@ufsc.br

G.G. Moreno
Department of Mechanical Engineering, University of Pamplona,
543050 Pamplona, Colombia

1 Introduction

All vehicles are subjected to high inertial forces when performing evasive maneuvers and turns. These forces directly influence vehicle stability, and, when a limit is reached, they may cause rollover. Rollover can be defined as any maneuver in which a vehicle rotates 90° or more around its longitudinal axis, and its side touches the ground.

The majority of existing lateral stability studies are planar, i.e. two-dimensional [1–3], considering that the vehicles have two contact points to the road when making a turn (the inner and outer tire on the turn). However, vehicles in general have at least four contact points that the authors believe have a significant influence on vehicle stability.

Three-dimensional vehicle analyses offer a new rollover insight, and, as consequence, an increase in road safety.

Davies' method, which is largely based on Graph theory, has proved to be very helpful in parallel robot kinestatic analysis, serial robots and vehicle mechanisms [4–7].

Our investigation focuses on the three-dimensional vehicle characteristics and the forces that influence its stability. We use Davies' method to describe the forces acting over the vehicle that can have influence on vehicle stability.

2 Static Analysis Tools

Static mechanism analyses can be made with three concepts: Screw theory, Graph theory and Davies' method [6–10]. These theories are explained in this section.

2.1 *Screw Theory*

This theory enables the mechanism to take an instantaneous position of representation in coordinate systems (successive screw displacement method) and the force and moment representation (wrench), replacing the traditional vector representation, as indicated below.

2.1.1 Method of Successive Screw Displacements

The screw displacement of a rigid body is represented by a rotation (θ) around an axis and a translation (d) along the same axis (screw axis— s) [4, 6, 10].

The bodies' instantaneous position (p_2) after screw displacement is given by the Eq. (1):

$$p_2 = Ap_1 \tag{1}$$

where (p_1) is the reference position matrix and $A_{4 \times 4}$ is the transformation matrix, which includes the rotation matrix, and the translation vector.

2.1.2 Wrench—Forces and Moments

In the static analysis, all mechanism forces and moments are represented by wrenches ($\A), as shown by Eq. (2) [8, 11],

$$\$^A = \left\{ \begin{matrix} s_{0i} \times s_i \\ s_i \end{matrix} \right\} F_i \quad \text{or} \quad \$^A = \left\{ \begin{matrix} s_i \\ 0 \end{matrix} \right\} M_i \tag{2}$$

where F_i is the force applied on joint i and M_i is the moment applied on joint i , s_i is the wrench i orientation vector, and $s_{oi} = [s_{oix} s_{oiy} s_{oiz}]^T$ is the instantaneous position vector of the joints and the center of gravity, related to the mechanism’s inertial reference point. In a more compact form, the wrench can be represented by Eq. (3)

$$\$^A = \hat{\$}^A \Psi \tag{3}$$

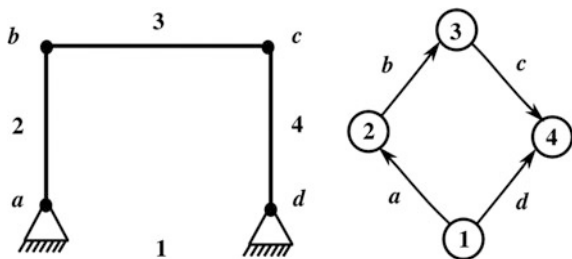
where $\hat{\A is the normalized wrench and Ψ its magnitude.

2.2 Graph Theory

Kinematic chains and mechanism are comprised of links and joints, which can be represented in a more abstract approach by graphs, where the vertices correspond to the links and the edges correspond to the joints [12].

Figure 1 illustrates the kinematic structure and the graph representation of a four-bar mechanism which contains four revolute joints “R” identified by the letters a, b, c and d , and four links identified by the numbers 1, 2, 3 and 4.

Fig. 1 Four-bar mechanism and the corresponding direct coupling graph



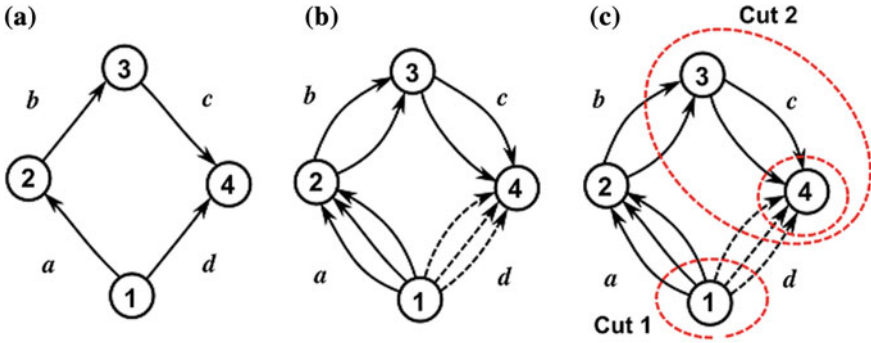


Fig. 2 **a** Direct coupling graph with branch and chord. **b** Direct coupling graph expanded. **c** Cut-set graph

The direct coupling graph can be represented by the incidence matrix $[I]$, as indicated in Eq. (4) [11, 12].

$$[I]_{4 \times 4} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & -1 \end{bmatrix}. \quad (4)$$

The incidence matrix provides the mechanism cut-set matrix $[Q]$ [11], Eq. (5), where each line represents a cut graph and the columns represent the mechanism joints. In addition, this matrix allows us to define three graph branches (edges a , b and c —identity matrix) and a chord (edge d), as shown in Fig. 2a.

$$[Q]_{3 \times 4} = \left[\begin{array}{ccc|c} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{array} \right] \quad (5)$$

For planar mechanism, revolute joint “R” allows a rotation and constrains two translations [12]. Figure 2b shows the direct coupling graph expanded with the constraints of each joint. Additionally, the external forces in the mechanism are also included: input torque (joint a) and output torque (joint d).

2.3 Davies Method for Statics

The Kirchhoff laws for electric circuits were adapted by Davies [13] to be used on mechanical systems.

Adapting the Kirchhoff-Davies cut-set law, it was possible to establish a relationship between actions belonging to the same partition or node, which contributes

to the static analysis. Davies states that the wrench algebraic sums belonging to the same partition or cut is zero, which is the Cut Law, as shown in Fig. 2c.

In the following sections, we make a classic vehicle stability analysis of a model in two dimensions; subsequently, we make a vehicle stability analysis of a new three-dimensional model using the proposed method.

3 Two-Dimensional Model of Vehicle Stability

Vehicle lateral stability can be evaluated through the Static Stability Factor (*SSF*). This factor is the maximum lateral acceleration— a_y (expressed in terms of gravity acceleration— g) in a quasi-static situation allowed before one or more of the tires lose contact with the ground [1, 2, 14].

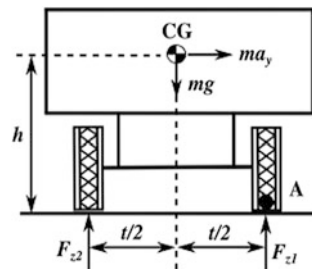
This coefficient estimation supposes a two-dimensional completely rigid vehicle supported by two tires, as presented in Fig. 3. During cornering, the lateral tire forces on the ground level (not shown) counterbalance the lateral inertial force acting on the vehicle gravity center, resulting in a roll moment. Considering the moments about the A point (Fig. 3), and at the rollover threshold condition, the normal load, F_{z2} reaches zero, then, the *SSF* factor can be calculated as shown in Eq. (6):

$$SSF = a_y/g = t/(2h) \tag{6}$$

4 Three-Dimensional Model of Vehicle Stability

Vehicle rollover is a three-dimensional phenomenon; affected by several vehicle characteristics, such as suspension, tires, chassis, gravity center height, vehicle track width, etc. The development and analysis of three-dimensional models enable a better understanding and interpretation of the rollover phenomenon, and allow the *SSF* factor estimation closer to reality. In addition, this analysis enables us to determine different vehicle characteristics influences on the rollover and a better understanding of the vehicle lateral load transfer.

Fig. 3 Vehicle geometric model



With this goal, and applying the proposed method, a vehicle model making a turn is developed and analyzed.

4.1 Kinematic Chain

The model considers a semi-rigid vehicle supported by four tires. The vehicle has independent movement of the front and the rear axle. At the start point, the vehicle has a planar constant curvilinear trajectory, starting from standstill with the velocity increasing by a constant acceleration, until reaching rollover threshold, as presented in Fig. 4a. First and fourth tires are named as outer tires; and second and third tires as inner tires in this curvilinear trajectory. The contact between the tire and the road presents three constraints to movement, as shown in Fig. 4b, [15, 16].

F_{xi} is the traction or brake force, F_{yi} is the lateral force and F_{zi} is the normal force. However at rollover threshold, tires 1 and 4 receive greater normal force than tires 2 and 3, and then they are not prone to slide laterally. We consider that tires 1 and 4 allow only vehicle rotation along the x -axis. Therefore, we model the tire contact with the road as pure revolute joint “R” along the x -axis. However, tires 2 and 3 have lateral deformation and may slide laterally, producing a track width change of their respective axles. For this reason, we model the tire contacts 2 and 3 as prismatic joints “P”. A three-dimensional vehicle model is presented in Fig. 5a.

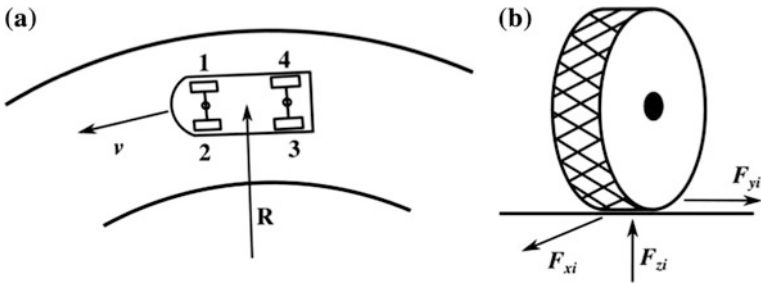


Fig. 4 a Vehicle model in a curvilinear trajectory. b Movement constraints in the tire-road contact

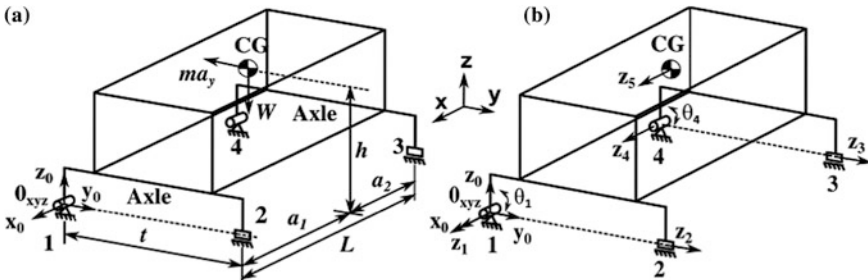


Fig. 5 The three-dimensional kinematic chain model and the mechanism position variables

Table 1 Screw parameters of the mechanism

Elements	s			s_o			θ	d
Joint 1	1	0	0	0	0	0	θ_1	0
Joint 2	0	1	0	0	0	0	0	t
Joint 3	0	1	0	$-L$	0	0	0	t
Joint 4	1	0	0	$-L$	0	0	θ_4	0
CG	1	0	0	$-a_1$	$t/2$	h	0	0

Table 2 Instantaneous position vector

Elements	s_{oi}		
Joint 1	0	0	0
Joint 2	0	$t \cos \theta_1$	$t \sin \theta_1$
Joint 3	$-L$	$t \cos \theta_4$	$t \sin \theta_4$
Joint 4	$-L$	0	0
CG	$-a_1$	$t/2$	h

4.2 Mechanism Position Kinematics

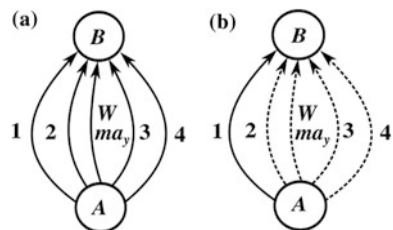
The mechanism’s kinematic model uses the successive screws method [9], shown in Sect. 2.1.1. Figure 5b and Table 1 present the screw parameters of the mechanism. Equation (1) enables us to determine the mechanism displacement and the instantaneous position vector $s_{oi} = [s_{oix}s_{oiy}s_{oiz}]^T$ of the joints and gravity center, as shown in Table 2.

where t is the vehicle track, L is wheel base, a_1 is distance between the front axle and the vehicle CG, h is the gravity center height, $\theta_{1,4}$ are the rotation angles of the revolute joints 1 and 4 respectively.

4.3 Direct Coupling Graph

As indicated in Sect. 2.2, a mechanism can be represented by a graph. Figure 6a shows the direct coupling graph that represents the mechanism shown in Fig. 5a.

Fig. 6 **a** Direct coupling graph. **b** Direct coupling graph with branches and chord



The graph has two vertices (A—road and B—vehicle’s body) and four edges representing the tire support (revolute joints 1 and 4 represent the outer tire supports, prismatic joints 2 and 3 are the inner tire supports, and the external forces (weight (W) and inertial force (ma_y)).

The direct coupling graph can be represented by the incidence matrix $[I]$, as indicated in Eq. (7) [11]:

$$I = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 & -1 & -1 \end{bmatrix} \quad (7)$$

The incidence matrix provides the mechanism cut-set matrix $[Q]$ [11], Eq. (8), where each line represents a cut graph and the columns represent the mechanism joints. In addition, this matrix allows us to define a branch graph (edge 1—identity matrix) and five chords (edges 2, 3, 4 and the external forces) as shown in Fig. 6b.

$$Q = [1 \quad | \quad 1 \quad 1 \quad 1 \quad 1 \quad 1] \quad (8)$$

4.4 The Mechanism Statics

Considering a static analysis in three-dimensional space [12]:

- (a) A revolute joint “ R ” allows a rotation and constrains two rotations and three translations. In the 1st and 4th joints, rotation around the x -axis is permitted, while rotations on the y and z -axis (M_y and M_z) and all three axes F_{xb} , F_{yi} and F_{zi} translations, are constrained.
- (b) A prismatic joint “ P ” allows two translations and three rotations, while constraining a translation. The translation on the z -axis (F_{zi}) is constrained for the 2nd and 3rd tires (joints).

All the constraints are represented as edges. This allows a cut-set graph amplification (Fig. 6b) and a cut-set matrix Eq. (8). Additionally, the mechanism external forces are also included, such as the vehicle weight (W) and the inertial force acting on the mechanism (ma_y).

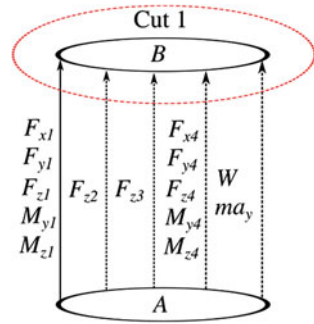
Figure 7 presents the cut-set action graph, and Eq. (9) presents the expanded cut-set matrix, where each line represents a cut graph, and the columns represent the joint constraints and the external forces present on the mechanism.

$$[Q]_{1 \times 14} = [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1] \quad (9)$$

The corresponding wrenches of each joint and external forces are defined according to Eq. (2), and Appendix A parameters.

All the mechanism wrenches compose the action matrix $[A_d]$ presented by Eq. (10).

Fig. 7 Cut-set action graph



$$[A_d]_{6 \times 14} = [\$^A F_{x1} \quad \$^A F_{y1} \quad \dots \quad \$^A W \quad \$^A ma_y] \quad (10)$$

According to Eq. (3), the wrench can be represented by a normalized wrench and a magnitude, therefore, from the Eq. (10) is obtained the unit action matrix $[\hat{A}_d]$ and the magnitude action vector $[\Psi]$, as presented by Eq. (11).

$$[\hat{A}_d]_{6 \times 14} = \left[\begin{matrix} \hat{\$}_{F_{x1}}^A & \hat{\$}_{F_{y1}}^A & \dots & \hat{\$}_W^A & \hat{\$}_{ma_y}^A \end{matrix} \right] \quad (11)$$

$$[\Psi]_{14 \times 1} = [F_{x1} \quad F_{y1} \quad \dots \quad W \quad ma_y]^T$$

4.5 Equation System Solution

Using the Cut Law [13], the algebraic sum of the normalized wrenches Eq. (11) that belong to the same cut (Fig. 7 and Eq. (9)) must be equal to zero.

Therefore, from Eqs. (9 and 11), the equation systems for the mechanism statics are defined, as shown in Eq. (12) (Appendix B):

$$[\hat{A}_n]_{6 \times 14} \cdot [\Psi]_{14 \times 1}^T = [0]_{6 \times 1} \quad (12)$$

where $[\hat{A}_n]$ is the network unit action matrix, and $[\Psi]$ is the action vector of the magnitudes of the mechanism. From the Eq. (12) system, it is necessary to identify the set of primary variables $[\Psi_p]$ (known variables), among the variables of Ψ . Once identified, the system is divided in two sets, as shown by Eq. (13),

$$[\hat{A}_{ns}]_{6 \times 12} [\Psi_s]_{12 \times 1}^T + [\hat{A}_{np}]_{6 \times 2} [\Psi_p]_{2 \times 1}^T = [0]_{6 \times 1} \quad (13)$$

where $[\Psi_p]$ is the vector of primary variables (known variables), $[\Psi_s]$ is the vector of secondary variables (unknown variables) (Eq. (14)), $[\hat{A}_{np}]$ are the columns corresponding to the primary variables and $[\hat{A}_{ns}]$ are the columns corresponding to the secondary variables.

$$\begin{aligned} [\Psi_p]_{2 \times 1} &= [W \quad ma_y]^T \\ [\Psi_s]_{12 \times 1} &= [F_{x1} \quad F_{y1} \quad \cdots \quad M_{y4} \quad M_{z4}]^T \end{aligned} \quad (14)$$

Solving the system in Eq. (13), provides

$$[\Psi_s]_{12 \times 1}^T = [\hat{A}_{ns}]_{6 \times 12}^{-1} [\hat{A}_{np}]_{6 \times 2} [\Psi_p]_{2 \times 1}^T \quad (15)$$

from which, the solution of the secondary variables $[\Psi_s]$ are expressed as a function of the primary variables $[\Psi_p]$.

5 Result Analysis

On the sixth row of Appendix B of the equation systems, we obtain the following equation, which relates the vehicle weight ($W = mg$), the inertial force acting on the mechanism (ma_y) and inner tire normal forces in the turn (F_{z2} and F_{z3}).

$$F_{z2} + F_{z3} + (h/t) m a_y - (1/2) m g = 0 \quad (16)$$

where h is the gravity center height, t is the vehicle track, m is the mass of vehicle, and a_y is the lateral acceleration measured on the vehicle's gravity center.

According to the static redundancy problem known as a "Four-legged Table" [17], a plane is defined by three points in space, and by consequence, a four-legged table has support plane multiplicities. This is why, when one leg loses contact with the ground, the table is supported by the other three.

Applying this theory in vehicle stability, and considering the chassis flexibility, suspension and tires stiffness; when a vehicle makes a turn, it is submitted to an increasing lateral load till it reaches the rollover threshold. During the maneuver, the rear inner tire (3) loses its contact to the ground. For this condition ($F_{z3} = 0$), and back to Eq. (16) one can get the static stability factor for a rigid vehicle as shown by Eq. (17):

$$SSF_{3D} = \frac{a_y}{g} = \frac{t}{2h} \left(1 - \frac{2F_{z2}}{W} \right) \quad (17)$$

This important theory is proved and observed in dynamic rollover tests, where it is possible to observe that the rear inner tire loses contact with the ground.

This information demonstrates that the SSF_{3D} factor of a vehicle Eq. (17) is in general inferior to the SSF_{2D} factor of a vehicle Eq. (6), because the second equation term is positive and lower than one.

It is possible to obtain a better vehicle stability representation and the SSF factor value attainments closer to the reality by means of Eq. (17).

In addition, in Eq. (17) it is shown that the SSF factor depends on the vehicle's weight and the normal force on the inner front tire F_{z2} as well.

6 Conclusions

This work has allowed us to demonstrate the Graph theory and Davies method versatilities in the mechanism analysis, achieving good results.

When the vehicle length is considered, the SSF factor becomes smaller, and this is a real problem. If we use the SSF_{2D} factor as an important feature to characterize the vehicle stability, we neglect the longitudinal effects. This means that the gravity center longitudinal position (a_l) and the LLT_A coefficient have important roles on the calculation of the SSF factor of the vehicle, and not only track (t) and gravity center height (h).

The theory about vehicle support points is possible due to different vehicle characteristics, such as suspension and tire stiffness, and chassis flexibility. This important analysis allows development of more complex models, in different types of vehicles, allowing better rollover phenomenon representation.

The SSF factor decrease is important, as it allows us to determine new road speed limits, which contributes to road safety and decreases vehicle stability related to accidents, which are very high nowadays.

Acknowledgments This research was supported by the Brazilian governmental agencies Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) and Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

Appendix A. Wrench Parameters of the Mechanism

Constraints and forces	s_i	s_{oi}		
F_{x1}	1 0 0	0	0	0
F_{y1}	0 1 0	0	0	0
F_{z1}	0 0 1	0	0	0
M_{y1}	0 1 0	0	0	0
M_{z1}	0 0 1	0	0	0
F_{z2}	0 0 1	0	$t \cos \theta_1$	$t \sin \theta_1$
F_{z3}	0 0 1	$-L$	$t \cos \theta_4$	$t \sin \theta_4$
F_{x4}	1 0 0	$-L$	0	0
F_{y4}	0 1 0	$-L$	0	0
F_{z4}	0 0 1	$-L$	0	0
M_{y4}	0 1 0	$-L$	0	0
M_{z4}	0 0 1	$-L$	0	0
W	0 0 -1	$-a_l$	$t/2$	h
$m a_y$	0 -1 0	$-a_l$	$t/2$	h

Appendix B. The Equations System from the Statics of the Mechanism

$$\begin{bmatrix}
 0 & 0 & 0 & 0 & 0 & t \cos \theta_1 & t \cos \theta_4 & 0 & 0 & 0 & 0 & 0 & | & -t/2 & h \\
 0 & 0 & 0 & 1 & 0 & 0 & L & 0 & 0 & L & 1 & 0 & | & -a_1 & 0 \\
 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -L & 0 & 0 & 1 & | & 0 & a_1 \\
 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & | & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & | & 0 & -1 \\
 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & | & -1 & 0
 \end{bmatrix} \cdot \begin{bmatrix}
 F_{x1} \\
 F_{y1} \\
 F_{z1} \\
 M_{y1} \\
 M_{z1} \\
 F_{z2} \\
 F_{z3} \\
 F_{x4} \\
 F_{y4} \\
 F_{z4} \\
 M_{y4} \\
 M_{z4} \\
 - \\
 W \\
 ma_y
 \end{bmatrix} = [0]_{6 \times 1}$$

References

1. Gillespie, T.D.: Fundamentals of vehicle dynamics. In: SAE International, 7th ed., ISBN1560911999 (1992)
2. Hac, A.: Rollover stability index including effects of suspension design. In: SAE International, SAE 2002 World Congress. Detroit, March 4-7 (2002)
3. Winkler, C.: Rollover of heavy commercial vehicles. In: UMTRI Research Review. The University of Michigan Transportation Research Institute, vol. 31, issue no. 4, pp. 1–20 (2000)
4. Lee, U.: A study on a method for predicting the vehicle controllability and stability using the screw axis theory. Ph.D. thesis. Hanyang University. Seoul, South Korea (2001)
5. Bidaud, P., Benamar, F., Poulain, T.: Kineto-static Analysis of an Articulated Six-wheel Rover. Climbing and Walking Robots, pp. 475–484. Springer Link (2006)
6. Erthal, J.: Modelo Cinestático para Análise de Rolagem em Veículos. Ph.D. These. Federal University of Santa Catarina. Florianópolis, Brazil (2010)
7. Mejia, L., Simas, H., Martins, D.: Force Capability Maximization of a 3RRR Symmetric Parallel Manipulator by Topology Optimization. 22nd International Congress of Mechanical Engineering (COBEM 2013), November 3–7, Ribeirão Preto, SP, Brazil (2013)
8. Davies, T. H.: The 1887 committee meets again. Subject: freedom and constraint. In: Ball 2000 Conference. Cambridge: Cambridge University Press, 56 (2000)
9. Tsai, L.-W.: Robot Analysis: the Mechanics of Serial and Parallel Manipulators. Wiley, New York. ISBN 0-471-32593-7 (1999)
10. Cazangi, H. R.: Aplicação do Método de Davies para Análise Cinemática e Estática de Mecanismos com Múltiplos Graus de Liberdade. Master's thesis. Federal University of Santa Catarina. Florianópolis, Brazil (2008)
11. Davies, T.H.: Coupling, coupling networks and their graphs. Mech. Mach. Theory **30**(7), 1001–1012 (1995)
12. Tsai, L.-W.: Mechanism Design: Enumeration of Kinematic Structures According to Function. London, CRC Press. ISBN 0849309018 (2001)

13. Davies, T.H.: Kirchhoff's circulation law applied to multi-loop kinematic chains. *Mech. Mach. Theory* **16**(3), 171–183 (1981)
14. Rill, G.: *Road Vehicle Dynamics: Fundamentals and Modeling*. CRC Press, ISBN 978-1-4398-3898-3 (2011)
15. Jazar, R.: *Vehicle Dynamics: Theory and Application*. 2nd edn. Springer, ISBN 978-1-4614-8544-5 (2014)
16. Pacejka, H.: *Tire and Vehicle Dynamics*. Published by Elsevier Ltd, 3rd ed. ISBN 9780080970165 (1995)
17. Heyman, J.: *Basic Structural Theory*. Cambridge: Cambridge University Press, 1st edn., ISBN-13 978-0-511-39692-2 (2008)

Analysis of the Kinematics of Planar Link Mechanism with Non-stationary Motion of Crank

J. Drewniak, P. Garlicka and B. Borowik

Abstract The article presents an example of using contour graphs to analyse the velocities and accelerations of components of the complex link mechanism without the use of decomposition of mechanism approach. The active link mechanism carries out the non-stationary motion, i.e. starting, stationary motion and stopping braking. The resulting system of algebraic linear equations solved for a given distribution of velocity and acceleration of the active link mechanism has been derived using the contour graph assigned to the considered mechanism.

Keywords Link mechanism · Contour graph without decomposition of mechanism · Kinematics · Non-stationary motion

1 Introduction

The classic analytical method of determination of velocities and accelerations of the links of particular link mechanisms consists in single differentiation of radiuses-vectors of positions of the mechanisms nodes [1–3]. Differentiation is performed according to a time variable. In the present paper, the methodology of application of the method of contour graphs is described. The methodology is utilized for performance of the kinematical analysis of a planar crane mechanism [4–8]. Contrary to the analytical method, the approach based on contour graphs leads to generation algebraic equations, only. The simplification of the analysis method was achieved due to omission of the step mechanism decomposition which was replaced via operations on the graph model [5, 8]. In general, the method of contour graphs can be utilized for both types of mechanisms i.e. planar and spatial ones, however their calculation model has to be considered as a closed kinematical chain [4].

J. Drewniak (✉) · P. Garlicka · B. Borowik
Faculty of Mechanical Engineering and Computer Science,
University of Bielsko-Biała, Bielsko-Biała, Poland
e-mail: jdrewniak@ath.bielsko.pl

The calculations performed by means of the method of contour graphs, without performance of system decomposition, are carried out according to the following roughly described algorithm:

- (a) schematic drawing of the analyzed mechanism, assuming the co-ordinate system $x0y$ as well as introductory determination of the geometric range of movements of nodes of the considered mechanism,
- (b) drawing the graph-based kinematical scheme of the mechanism having N independent contours,
- (c) generation of the system of vector equations for the velocities for every complete closed contour (N closed cycles, circuits—according to the utilized style of nomenclature) together with its solution (after appropriate recalculations and solving),
- (d) generation of the system of vector equations for the accelerations for every complete contour (N closed cycles, circuits—according to the utilized style of nomenclature) together with its solution (after appropriate recalculations and solving).

2 Equations of the Contour Graph Method [4]

For the closed kinematical chain consisting of n movable links and one fixed base 0 , we can write the first vector equation of the contour graph methods for every closed kinematical chain:

$$\omega_{1,0} + \omega_{2,1} + \dots + \omega_{0,n} = \mathbf{0}, \quad (1)$$

which can be written in the following form (2):

$$\sum_i \omega_{i,i-1} = \mathbf{0}, \quad (2)$$

where $\omega_{i,i-1}$ —relative rotational velocity of the link i in relation to the link $i - 1$.

The second vector equation of the contour graph method can be derived from the second basic kinematical relationship related to the reciprocating-rotational motion i.e. planar movement. It is presented in the abbreviated version by the formula (3). It is also written for every contour. Therefore, it is considered as a second vector equation of the contour graph method. These two types of equations allow for determination of the angular velocities $\omega_{i,i-1}$:

$$\sum_i \mathbf{r}_{Ai} \times \omega_{i,i-1} + \sum_i \mathbf{v}_{Ai,i-1}^r = \mathbf{0}. \quad (3)$$

Analysis of the accelerations of mechanism links by means of the contour graph method is based—similarly to the method of analysis of velocities—on the basic kinematical equation related to accelerations of rigid bodies:

$$\mathbf{a}_{A_j} = \mathbf{a}_{A_k} + \mathbf{a}_{A_j,k}^r + \mathbf{a}_{A_j,k}^c, \tag{4}$$

where \mathbf{a}_{A_j} , \mathbf{a}_{A_k} —absolute linear accelerations of the same point \mathbf{A} , considered firstly as belonging to the link j , then second time—as belonging to the link k —therefore: $A_j \in j$ and $A_k \in k$; $\mathbf{a}_{A_j,k}^r \equiv \mathbf{a}_{A_j,A_k}^r$ —relative acceleration of the point $A_j \in j$ in relation to the point $A_k \in k$; $\mathbf{a}_{A_j,k}^c \equiv \mathbf{a}_{A_j,A_k}^c$ —Coriolis acceleration of the point $A_j \in j$ in relation to the point $A_k \in k$, where $\mathbf{a}_{A_j,k}^c = 2 \cdot \boldsymbol{\omega}_k \times \mathbf{v}_{A_j,k}^r$.

The absolute angular acceleration $\boldsymbol{\varepsilon}_i = \boldsymbol{\varepsilon}_{i,0}$ of the rigid link is expressed by the following formula:

$$\boldsymbol{\varepsilon}_i = \boldsymbol{\varepsilon}_{i-1} + \boldsymbol{\varepsilon}_{i,i-1} + \boldsymbol{\omega}_i \times \boldsymbol{\omega}_{i-1}, \tag{5}$$

where $\boldsymbol{\varepsilon}_{i-1} = \boldsymbol{\varepsilon}_{i-1,0}$ is the absolute angular acceleration of the link $i - 1$ and $\boldsymbol{\varepsilon}_{i,i-1}$ is the absolute angular acceleration of the link i in relation to the link $i - 1$.

After rewriting of Eq. (5) into r equations (separately for each link) and summation of these equations, one can obtain the first vector equation for angular accelerations. According to the contour graph method, such an equation can be written for every contour:

$$\sum_i \boldsymbol{\varepsilon}_{i,i-1} + \sum_i \boldsymbol{\omega}_i \times \boldsymbol{\omega}_{i-1} = \mathbf{0}. \tag{6}$$

The linear acceleration of the point A_i belonging to the link i is equal:

$$\mathbf{a}_{A_i,i} = \mathbf{a}_{A_i,i-1} + \mathbf{a}_{A_i,i-1}^r + \mathbf{a}_{A_i,i-1}^c \tag{7}$$

where $\mathbf{a}_{A_i,i-1}$ is an acceleration of the point A_i belonging to the link $i - 1$, $\mathbf{a}_{A_i,i-1}^r \equiv \mathbf{a}_{A_i,i-1}^r$ —is the relative acceleration of the point A_i belonging to the link i in relation to the same point A_i but considered as belonging to the link $i - 1$ and $\mathbf{a}_{A_i,i-1}^c$ is the Coriolis acceleration defined as:

$$\mathbf{a}_{A_i,i-1}^c = 2 \cdot \boldsymbol{\omega}_{i-1} \times \mathbf{v}_{A_i,i-1}^r.$$

The relationship for the linear acceleration $\mathbf{a}_{A_{i+1},i}$ of the point $A_{i+1} \in i$ considering a dependence on the linear acceleration $\mathbf{a}_{A_i,i}$ of the point $A_{i+1} \in i$ (i.e.: for two separate points A_{i+1} and A_i belonging to the same link i) is expressed in the following form:

$$\mathbf{a}_{A_{i+1},i} = \mathbf{a}_{A_i,i} + \boldsymbol{\varepsilon}_i \times \mathbf{r}_{A_i A_{i+1}} + \boldsymbol{\omega}_i \times (\boldsymbol{\omega}_i \times \mathbf{r}_{A_i A_{i+1}}), \quad (8)$$

where $\boldsymbol{\varepsilon}_i = \boldsymbol{\varepsilon}_{i,0}$ is the absolute angular acceleration of the link i , whereas $\mathbf{r}_{A_i A_{i+1}}$ is the vector-radius between the discussed points A_i and A_{i+1} .

Inserting the relationship (7) into the formula (8), we obtain a formula for determination of the linear acceleration of the point $A_{i+1,i}$, so the point A_{i+1} belonging to the link i :

$$\mathbf{a}_{A_{i+1},i} = \mathbf{a}_{A_{i,i-1}} + \mathbf{a}_{A_{i,i-1}}^r + \mathbf{a}_{A_{i,i-1}}^c + \boldsymbol{\varepsilon}_i \times \mathbf{r}_{A_i A_{i+1}} + \boldsymbol{\omega}_i \times (\boldsymbol{\omega}_i \times \mathbf{r}_{A_i A_{i+1}}). \quad (9)$$

Further activities are as follows: inserting of the above equation into an equation enclosing all links of the considered mechanism, summation of these equations and inserting of the relationship for the vector-radius $\mathbf{r}_{A_{i-1} A_i}$ (10):

$$\mathbf{r}_{A_{i-1} A_i} = \mathbf{r}_{A_i} - \mathbf{r}_{A_{i-1}}, \quad (10)$$

where $\mathbf{r}_{A_i} = \mathbf{r}_{O A_i}$ and $\mathbf{r}_{A_{i-1}} = \mathbf{r}_{O A_{i-1}}$.

Finally, we obtain the second vector equation of the method of contour graphs for determination of angular accelerations $\boldsymbol{\varepsilon}_{i,i-1}$:

$$\begin{aligned} & \left[\mathbf{a}_{A_{1,0}}^r + \mathbf{a}_{A_{2,1}}^r + \cdots + \mathbf{a}_{A_{i,i-1}}^r + \cdots + \mathbf{a}_{A_{0,n}}^r \right] \\ & + \left[\mathbf{a}_{A_{1,0}}^c + \mathbf{a}_{A_{2,1}}^c + \cdots + \mathbf{a}_{A_{i,i-1}}^c + \cdots + \mathbf{a}_{A_{0,n}}^c \right] \\ & + \left[\mathbf{r}_{A_1} \times (\boldsymbol{\varepsilon}_{1,0} + \boldsymbol{\omega}_1 \times \boldsymbol{\omega}_{1,0}) + \cdots + \mathbf{r}_{A_i} \times (\boldsymbol{\varepsilon}_{i,i-1} + \boldsymbol{\omega}_i \times \boldsymbol{\omega}_{i,i-1}) \right. \\ & + \cdots + \mathbf{r}_{A_0} \times (\boldsymbol{\varepsilon}_{0,n} + \boldsymbol{\omega}_0 \times \boldsymbol{\omega}_{0,n}) \left. \right] \\ & + \left[\boldsymbol{\omega}_1 \times (\boldsymbol{\omega}_1 \times \mathbf{r}_{A_1 A_2}) + \cdots + \boldsymbol{\omega}_i \times (\boldsymbol{\omega}_i \times \mathbf{r}_{A_i A_{i+1}}) \right. \\ & + \cdots + \boldsymbol{\omega}_0 \times (\boldsymbol{\omega}_0 \times \mathbf{r}_{A_0 A_1}) \left. \right] = \mathbf{0} \end{aligned} \quad (11)$$

which can be written in the abbreviated form (12):

$$\begin{aligned} & \sum_i \mathbf{a}_{A_{i,i-1}}^r + \sum_i \mathbf{a}_{A_{i,i-1}}^c + \sum_i \mathbf{r}_{A_i} \times (\boldsymbol{\varepsilon}_{i,i-1} \times \boldsymbol{\omega}_i - \boldsymbol{\omega}_{i,i-1}) \\ & + \sum_i \mathbf{r}_{A_i} \times (\boldsymbol{\varepsilon}_{i,i-1} + \boldsymbol{\omega}_i \times \boldsymbol{\omega}_{i,i-1}) + \sum_i \boldsymbol{\omega}_i \times (\boldsymbol{\omega}_i \times \mathbf{r}_{A_i A_{i+1}}) = \mathbf{0} \end{aligned} \quad (12)$$

3 Problem Description

The analysis of the link mechanism is performed utilizing the contour graph methodology. The detailed considerations are presented below. The elements of the system are e.g.: crank 1 which is considered as a driving link and the slider 5 which is considered as a driven link (Fig. 1). The geometrical data related to the lengths of

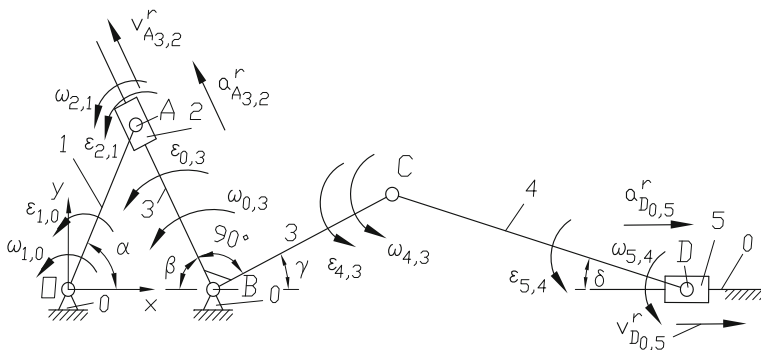


Fig. 1 Analyzed mechanism, the angles, velocities, accelerations and links are designated

the links were assumed as follows: $|OA| = 80$, $|OB| = 60$, $|BC| = 120$, $|CD| = 250$. The range of rotational movement of the crank 1: $\alpha \in (38^\circ; 128^\circ)$, whereas for angles $\alpha \in (38^\circ; 50^\circ)$ —startup (uniformly accelerated motion), for angles $\alpha \in (50^\circ; 116^\circ)$ —uniform motion, for angles $\alpha \in (116^\circ; 128^\circ)$ —braking (uniformly decelerated motion). In Fig. 1, the scheme of the mechanism is presented and all descriptions of the links and geometrical quantities are named, as well.

Charts of distributions of velocities and accelerations are presented in Figs. 2 and 3.

Fig. 2 Chart of the angular velocity of the crank 1 as a function of its rotation angle

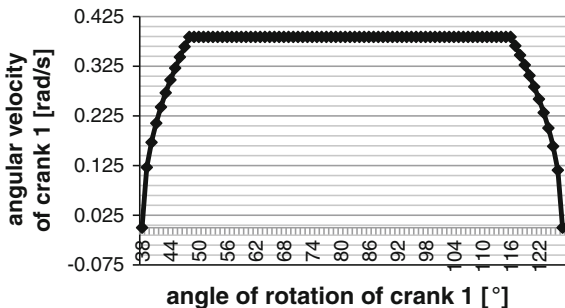
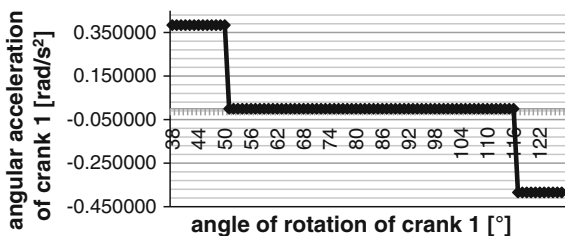


Fig. 3 Chart of angular acceleration of the crank 1 as a function of its rotation angle



4 Derivation of the Vector Equation System for Velocities

The problem of kinematical analysis of the closed link mechanisms is solved in the present paper without decomposition of the considered system.

The maximum number of independent contours is given by

$$N = c - n = 7 - 5 = 2,$$

where c is the number of joints ($c = 7$), and n is the number of moving links ($n = 5$).

Therefore, the two-contour graph-based kinematical scheme of the analyzed mechanism is considered. It is presented in Fig. 4.

The introductory equations of the discussed method are the vector equations obtained based upon gathering of two system of equations assigned to two graph contours I and II [4–6]:

$$\boldsymbol{\omega}_{1,0} + \boldsymbol{\omega}_{2,1} + \boldsymbol{\omega}_{0,3} = \mathbf{0} \quad (13a)$$

$$\mathbf{r}_{OA} \times \boldsymbol{\omega}_{2,1} + \mathbf{r}_{OB} \times \boldsymbol{\omega}_{0,3} + \mathbf{v}_{A3,2}^r = \mathbf{0} \quad (13b)$$

$$\boldsymbol{\omega}_{3,0} + \boldsymbol{\omega}_{4,3} + \boldsymbol{\omega}_{5,4} = \mathbf{0} \quad (13c)$$

$$\mathbf{r}_{OB} \times \boldsymbol{\omega}_{3,0} + \mathbf{r}_{OC} \times \boldsymbol{\omega}_{4,3} + \mathbf{r}_{OD} \times \boldsymbol{\omega}_{5,4} + \mathbf{v}_{D0,5}^r = \mathbf{0}. \quad (13d)$$

The arbitrary angular velocities of crank 1 of the mechanism $\boldsymbol{\omega}_{1,0} = \omega_{1,0} \cdot \mathbf{k}$ is depicted in Fig. 2.

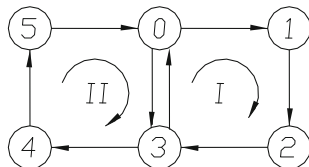
We search for the angular velocities of the remaining linkages (according to Fig. 1):

$$\begin{aligned} \boldsymbol{\omega}_{2,1} &= \omega_{2,1} \cdot \mathbf{k}, & \boldsymbol{\omega}_{0,3} &= \omega_{0,3} \cdot \mathbf{k}, & \boldsymbol{\omega}_{3,0} &= -\omega_{3,0} \cdot \mathbf{k}, \\ \boldsymbol{\omega}_{0,3} &= \omega_{3,0}, & \boldsymbol{\omega}_{4,3} &= \omega_{4,3} \cdot \mathbf{k}, & \boldsymbol{\omega}_{5,4} &= \omega_{5,4} \cdot \mathbf{k}. \end{aligned}$$

Moreover, we search for the linear velocities of the sliders 2 and 5:

$$\mathbf{v}_{A3,2}^r = v_{A3,2x}^r \cdot \mathbf{i} + v_{A3,2y}^r \cdot \mathbf{j} = -v_{A3,2}^r \cdot \cos \beta \cdot \mathbf{i} + v_{A3,2}^r \cdot \sin \beta \cdot \mathbf{j}$$

Fig. 4 Two-contour graph-based kinematical scheme of the analyzed mechanism



and

$$\mathbf{v}_{D0,5}^r = v_{D0,5x}^r \cdot \mathbf{i} = v_{D0,5}^r \cdot \mathbf{i} = -v_{D5,0x}^r \cdot \mathbf{i} = -v_{D5,0}^r \cdot \mathbf{i}.$$

The vector products in Eq. (13b) and (13d) can be rearranged to the following form (according to Fig. 1):

$$\begin{aligned} \mathbf{r}_{OA} \times \boldsymbol{\omega}_{2,1} &= y_A \cdot \omega_{2,1} \cdot \mathbf{i} - x_A \cdot \omega_{2,1} \cdot \mathbf{j}, \\ \mathbf{r}_{OB} \times \boldsymbol{\omega}_{0,3} &= y_B \cdot \omega_{0,3} \cdot \mathbf{i} - x_B \cdot \omega_{0,3} \cdot \mathbf{j}, \\ \mathbf{r}_{OB} \times \boldsymbol{\omega}_{3,0} &= y_B \cdot (-\omega_{3,0}) \cdot \mathbf{i} - x_B \cdot (-\omega_{3,0}) \cdot \mathbf{j}, \\ \mathbf{r}_{OC} \times \boldsymbol{\omega}_{4,3} &= y_C \cdot \omega_{4,3} \cdot \mathbf{i} - x_C \cdot \omega_{4,3} \cdot \mathbf{j}, \\ \mathbf{r}_{OD} \times \boldsymbol{\omega}_{5,4} &= y_D \cdot \omega_{5,4} \cdot \mathbf{i} - x_D \cdot \omega_{5,4} \cdot \mathbf{j} \end{aligned}$$

The next step consists in projection of equations (13) on the adequate axes of the Cartesian system \mathbf{Oxyz} . We then obtain the system of algebraic equations:

$$\omega_{1,0} + \omega_{2,1} + \omega_{0,3} = 0 \quad \text{projection on the axis } z \quad (14a)$$

$$-\omega_{3,0} + \omega_{4,3} + \omega_{5,4} = 0 \quad \text{projection on the axis } z \quad (14b)$$

$$y_A \cdot \omega_{2,1} + y_B \cdot \omega_{3,0} - v_{A3,2}^r \cdot \cos \beta = 0 \quad \text{projection on the axis } x \quad (14c)$$

$$-x_A \cdot \omega_{2,1} - x_B \cdot \omega_{3,0} + v_{A3,2}^r \cdot \sin \beta = 0 \quad \text{projection on the axis } y \quad (14d)$$

$$-y_B \cdot \omega_{3,0} + y_C \cdot \omega_{4,3} + y_D \cdot \omega_{5,4} + v_{D0,5}^r = 0 \quad \text{projection on the axis } x \quad (14e)$$

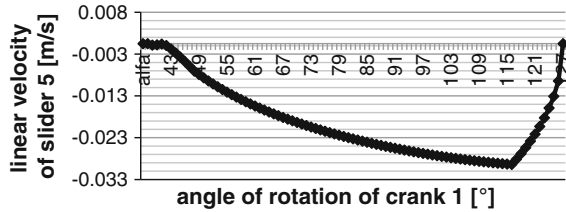
$$x_B \cdot \omega_{3,0} - x_C \cdot \omega_{4,3} - x_D \cdot \omega_{5,4} = 0 \quad \text{projection on the axis } y. \quad (14f)$$

Finally, we obtain a system of six algebraic equations enclosing six unknown quantities: $\omega_{2,1}$, $\omega_{3,0} = \omega_{0,3}$, $\omega_{4,3}$, $\omega_{5,4}$, $v_{A3,2}^r$ and $v_{D5,0}^r$.

5 Generation of a System of Vector Equations for the Accelerations of Mechanism Links

The task of an analysis of the accelerations of closed links mechanisms will be also solved without an application of decomposition of the considered system. We use here the same two-contour graph being the kinematical model of the analyzed mechanism presented in Fig. 4. The introductory equations of the discussed method are the vector equations obtained based upon the union of two systems of equations dedicated to both parts of the considered mechanism [4, 5, 7]:

Fig. 5 Chart of linear velocity of the slider 5 as a function of the rotational angle of the crank 1



$$\boldsymbol{\varepsilon}_{1,0} + \boldsymbol{\varepsilon}_{2,1} + \boldsymbol{\varepsilon}_{0,3} = \mathbf{0}, \quad (15a)$$

$$\mathbf{r}_{OA} \times \boldsymbol{\varepsilon}_{2,1} + \mathbf{r}_{OB} \times \boldsymbol{\varepsilon}_{0,3} + \mathbf{a}_{A3,2}^r + \mathbf{a}_{A3,2}^c - \omega_{1,0}^2 \cdot \mathbf{r}_{OA} - \omega_{0,3}^2 \cdot \mathbf{r}_{AB} = \mathbf{0}, \quad (15b)$$

$$\boldsymbol{\varepsilon}_{3,0} + \boldsymbol{\varepsilon}_{4,3} + \boldsymbol{\varepsilon}_{5,4} = \mathbf{0}, \quad (15c)$$

$$\begin{aligned} \mathbf{r}_{OB} \times \boldsymbol{\varepsilon}_{3,0} + \mathbf{r}_{OC} \times \boldsymbol{\varepsilon}_{4,3} + \mathbf{r}_{OD} \times \boldsymbol{\varepsilon}_{5,4} + \mathbf{a}_{D0,5}^r \\ + \mathbf{a}_{D0,5}^c - \omega_{3,0}^2 \cdot \mathbf{r}_{BC} - \omega_{4,0}^2 \cdot \mathbf{r}_{CD} = \mathbf{0}. \end{aligned} \quad (15d)$$

The velocities of all links and angular accelerations of the crank 1 of the mechanism (Figs. 2, 3 and 5) are known.

We are searching for the angular and linear accelerations of the remaining links (according to Fig. 1):

$$\begin{aligned} \boldsymbol{\varepsilon}_{2,1} = \varepsilon_{2,1} \cdot \mathbf{k}, \quad \boldsymbol{\varepsilon}_{0,3} = \varepsilon_{0,3} \cdot \mathbf{k}, \quad \boldsymbol{\varepsilon}_{3,0} = -\varepsilon_{3,0} \cdot \mathbf{k}, \quad \boldsymbol{\varepsilon}_{4,3} = \varepsilon_{4,3} \cdot \mathbf{k}, \quad \boldsymbol{\varepsilon}_{5,4} = \varepsilon_{5,4} \cdot \mathbf{k}, \\ \mathbf{a}_{A3,2}^r = a_{A3,2x}^r \cdot \mathbf{i} + a_{A3,2y}^r \cdot \mathbf{j} = -a_{A3,2}^r \cdot \cos \beta \cdot \mathbf{i} + a_{A3,2}^r \cdot \sin \beta \cdot \mathbf{j}, \end{aligned}$$

and

$$\mathbf{a}_{D0,5}^r = a_{D0,5x}^r \cdot \mathbf{i} = a_{D0,5}^r \cdot \mathbf{i} = -a_{D5,0x}^r \cdot \mathbf{i} = -a_{D5,0}^r \cdot \mathbf{i}.$$

The vector products in Eq. (15b) and (15d) can be converted into the following form:

$$\begin{aligned} \mathbf{r}_{OA} \times \boldsymbol{\varepsilon}_{2,1} &= y_A \cdot \varepsilon_{2,1} \cdot \mathbf{i} - x_A \cdot \varepsilon_{2,1} \cdot \mathbf{j}, \\ \mathbf{r}_{OB} \times \boldsymbol{\varepsilon}_{0,3} &= y_B \cdot \varepsilon_{0,3} \cdot \mathbf{i} - x_B \cdot \varepsilon_{0,3} \cdot \mathbf{j}, \\ \mathbf{r}_{OB} \times \boldsymbol{\varepsilon}_{3,0} &= y_B \cdot (-\varepsilon_{3,0}) \cdot \mathbf{i} - x_B \cdot (-\varepsilon_{3,0}) \cdot \mathbf{j}, \\ \mathbf{r}_{OC} \times \boldsymbol{\varepsilon}_{4,3} &= y_C \cdot \varepsilon_{4,3} \cdot \mathbf{i} - x_C \cdot \varepsilon_{4,3} \cdot \mathbf{j}, \\ \mathbf{r}_{OD} \times \boldsymbol{\varepsilon}_{5,4} &= y_D \cdot \varepsilon_{5,4} \cdot \mathbf{i} - x_D \cdot \varepsilon_{5,4} \cdot \mathbf{j}. \end{aligned}$$

The Coriolis acceleration $\mathbf{a}_{A3,2}^c$ of the point **A** on link 3 in relation to the point **A** on link 2 can be expressed in the following way:

$$\begin{aligned}
\mathbf{a}_{A3,2}^c &= 2 \cdot \omega_{3,0} \times \mathbf{v}_{A3,2}^r \\
&= 2 \cdot (-\omega_{3,0} \cdot \mathbf{k}) \times \left(-v_{A3,2x}^r \cdot \mathbf{i} + v_{A3,2y}^r \cdot \mathbf{j} \right) \\
&= 2 \cdot \omega_{3,0} \cdot \left(v_{A3,2}^r \cdot \sin \beta \cdot \mathbf{i} + v_{A3,2}^r \cdot \cos \beta \cdot \mathbf{j} \right) \\
&= 2 \cdot \omega_{3,0} \cdot v_{A3,2}^r \cdot \sin \beta \cdot \mathbf{i} + 2 \cdot \omega_{3,0} \cdot v_{A3,2}^r \cdot \cos \beta \cdot \mathbf{j}.
\end{aligned}$$

Then, we perform projection of Eq. (15a) onto the axes of the Cartesian system **Oxyz**. As a result, we obtain the following system of algebraic equations:

$$\varepsilon_{1,0} + \varepsilon_{2,1} - \varepsilon_{3,0} = 0, \quad (16a)$$

$$-\varepsilon_{3,0} + \varepsilon_{4,3} + \varepsilon_{5,4} = 0, \quad (16b)$$

$$\begin{aligned}
y_A \cdot \varepsilon_{2,1} - y_B \cdot \varepsilon_{3,0} - a_{A3,2}^r \cdot \cos \beta + 2 \cdot \omega_{3,0} \cdot v_{A3,2}^r \cdot \sin \beta \\
- \omega_{1,0}^2 \cdot x_A - \omega_{3,0}^2 \cdot (x_B - x_A) = 0,
\end{aligned} \quad (16c)$$

$$\begin{aligned}
-x_A \cdot \varepsilon_{2,1} - x_B \cdot \varepsilon_{3,0} + a_{A3,2}^r \cdot \sin \beta + 2 \cdot \omega_{3,0} \cdot v_{A3,2}^r \cdot \cos \beta \\
- \omega_{1,0}^2 \cdot y_A - \omega_{3,0}^2 \cdot (-y_A) = 0,
\end{aligned} \quad (16d)$$

$$-y_B \cdot \varepsilon_{3,0} + y_C \cdot \varepsilon_{4,3} + a_{D0,5}^r - \omega_{3,0}^2 \cdot (x_C - x_B) - \omega_{5,4}^2 \cdot (x_D - x_C) = 0 \quad (16e)$$

$$-x_B \cdot (-\varepsilon_{3,0}) - x_C \cdot \varepsilon_{4,3} - x_D \cdot \varepsilon_{5,4} - \omega_{3,0}^2 \cdot y_C - \omega_{5,4}^2 \cdot (-y_C) = 0. \quad (16f)$$

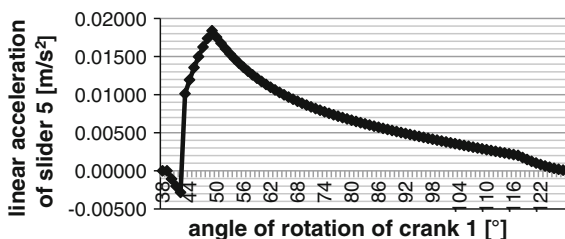
It is the system consisting of six equation enclosing six unknowns: $\varepsilon_{2,1}$, $\varepsilon_{3,0}$, $\varepsilon_{4,3}$, $\varepsilon_{5,4}$, $a_{A3,2}^r$ and $a_{D0,5}^r$.

6 Analysis of the Results

The systems of equations (14a) and (16a) have been solved utilizing a spreadsheet. The solutions are related to the whole range of working cycle of the mechanism i.e.: startup, steady motion and braking of the driving crank 1. The data for these special periods of working cycle are presented in charts (Figs. 5 and 6). All unknowns mentioned in Chaps. 4 and 5 were determined. The chart of linear velocity of the slider 5 (as a function of rotational angle of the crank 1) is presented in Fig. 5.

Moreover, in Fig. 6, the chart of the acceleration of the slider 5 as a function of the rotational angle of the crank 1 is presented. The same data were taken into account (Figs. 3 and 4) as well as the values of velocities of links—determined from the system of equations (16).

Fig. 6 Chart of linear acceleration of the slider 5 as a function of the rotational angle of the crank 1



7 Conclusions

Based upon the calculations performed in the present paper, it can be stated that the method of contour graphs is a very effective tool suitable for kinematic analysis of mechanisms built of rigid links. The advantages of this method are even more recognizable when we analyze motions of the mechanism within the full range of its work i.e. taking into account the periods of startup, steady motion, braking and reversion. Therefore, it has been proved that the method of contour graphs without decomposition of the analyzed mechanism into some more simple substructures is equally effective as the version with decomposition—presented in the monograph [4]. Additionally, due to the fact that the method of contour graphs utilizes nothing but just the algebraic equations, solving of the reverse problems of kinematics would be essentially made easier. Additionally, it should be emphasised that the method of contour graphs could be utilized for performance kinematical and dynamical analysis of spatial mechanisms.

References

1. Frączek, J., Wojtyra, M.: Kinematics of Multi-body Systems. Calculation Methods (in Polish). WNT, Warszawa (2008)
2. Adamczyk, E., Jucha, J., Miller, S.: Theory of Mechanisms and Machines (in Polish). Politechnika Wroclawska, Wrocław (1976)
3. Parszewski, Z.: Theory of Machines and Mechanisms (in Polish). WNT, Warszawa (1967)
4. Marghitu, D.B.: Kinematic Chains and Machine Components Design, 792 p. Elsevier, Amsterdam (2005)
5. Dobija, M.: Kinematical Analysis of Closed Crane Mechanisms by Means of the Contour Graph Method (in Polish). Engineer Diploma Work, University of Bielsko-Biala (2014)
6. Drewniak, J., and Garlicka, P.: Method of analysis of velocities of the crane mechanism (in Polish), Logistyka, 3 (2015)
7. Drewniak, J., Garlicka, P.: Method of analysis of accelerations of the crane mechanism (in Polish). Logistyka, 3 (2015)
8. Dobija, M., Drewniak, J., Kopeć, J., Zawiślak, S., Shingissov, B., Algazy, Z.: Contour graph application in kinematical analysis of crane mechanism. Współczesne trendy w teorii maszyn i układów mechatronicznych. Proceedings: XXIV Międzynarodowa Konferencja Naukowa Teorii Maszyn i Układów Mechatronicznych, Wrocław-Szklarska Poręba, 135 (2014)

Part III
Modeling of Production Processes

Petri Nets for Computer Aided Group Technology

R. Stryczek

Abstract An increasing number of various versions of Petri nets and the tools for modeling them raises a question: which variant is the most suitable? What are its capabilities and limitations for solving a particular problem? This work presents a chosen Petri net class as a tool for modeling group manufacturing processes, verifying their completeness and coherence as far as optimization of the machined parts flow through the manufacturing cell. A timed priority Petri net has been proposed. The modules of the manufacturing process model have been presented. A tested genetic algorithm with a chromosome in form of an ordered set of priorities of the chosen events has been used for the manufacturing cell performance scheduling. The work includes an example in the form of CNC lathe machining process.

Keywords Manufacturing process modeling · Group technology · Cellular manufacturing · Genetic algorithm

1 Petri Nets as a Formal Language for Conditions and Events System Specification

Nowadays, after several years of development and improvement, Petri nets methodology is a full-grown, widely popularized tool for modeling, analyzing, planning, flow control, simulation, scheduling of various and complex condition-event systems. Researchers have developed a wide range of Petri nets classes. There are timed, hierarchical, colored, boolean, fuzzy, attribute, priority, predicate, stochastic and hybrid Petri nets. At the same time, a huge amount of bundled software supporting net modeling by means of bipartite graphs such as a Petri net has been developed. A bipartite graph consists of two disjoint sets of nodes representing places (conditions) and transitions (events). The possibility of graphic

R. Stryczek (✉)

University of Bielsko-Biala, Bielsko-Biala, Poland

e-mail: rstryczek@ath.bielsko.pl

presentation is a huge benefit at the stage of planning and simulating. Nonetheless, we have to bear in mind that the actual models of manufacturing systems are very extensive, thus the need for modularization of a model expressed in block structure and/or hierarchical structure. Arcs in Petri nets are directed arcs divided into two subsets: the arcs of place \rightarrow transition type and the arcs of transition \rightarrow place type. Significant simplification of a model and easier planning can be achieved by introduction of an additional set of inhibitor arcs connecting places with transitions. A transition connected to a place by means of an inhibitor arc cannot fire when the place contains a marker.

A net proposed by Carl Adam Petri did not include a time function. It has been added in the following years and it is represented in the net in many ways [12]. Generally, we can point out two approaches: the time in which a marker remains in a particular place or the time in which the firing of transition occurs. Transitions, often called events, constitute dynamic elements of a net so the second approach seems more natural. From a practical point of view, time generally has a discrete form in a model, e.g., it is rounded to full seconds. Therefore, many system override events possess ascribed time equal to zero. Completing a model with a time function we gain the possibility to analyze time series in a project. Output data from model simulation are the basis for economical and organizational analysis, such as: efficiency, the use of working stations, and computing the length of the manufacturing cycle.

As far as planning and realizing of manufacturing processes is concerned, the usefulness, flexibility and universality of a manufacturing system model in the form of a Petri net have been noticed relatively quickly. Undoubtedly, in this field, a Petri net was used most often as a tool for modeling and control of the production process in Flexible Manufacturing Systems (FMS) [1, 12, 23]. The main functions of FSM are: deadlock prevention [21] and production scheduling [20]. The most developed class of Petri nets was used in the work [7] in order to solve the machine-loading problems and to maximize flow capacity in FMS. A new class of Petri net model called the "Extended neuro fuzzy Petri net" has been constructed to capture clearly the various details of the machine loading problem.

Computer-aided process planning (CAPP) has also taken advantage of the approach based on Petri nets model for generating processes structures and manufacturing process, resources use or retain manufacturing knowledge [2, 18]. First attempts at using Petri nets in manufacturing processes planning concerned mainly the issue of matching it with the conditions resulting from the potential of the production division. The proposition to use the data from the Petri net model which controlled the production process in order to verify the manufacturing process planning is worth mentioning, because as a result the CAPP dynamic system prototype was created [17]. Another team concentrated on the analysis of the possibility of operation sequencing in manufacturing process and on the presentation of alternative versions of manufacturing process based on Petri net model [5]. At the same time Lee and Jung [10] present the prognosis of Petri net use in two goals of modeling the knowledge related to the selection and sequencing of the steps of the process as far as the flexible representation of the steps sequence. The

complete approach was presented in work [16]. It included knowledge acquisition as well as process structure modeling and the generated Petri net evaluation. The authors aimed at building the methodology of manufacturing process modeling based on knowledge.

The most developed project in the field of modeling and analysis of manufacturing processes with the use of a Petri net is Process Specification Language (PSL). The work on this project was coordinated by the National Institute of Standards and Technology (NIST) USA. The technical representation of the PSL concept for manufacturing process planning is a Petri net class called Compact Process Planning net (CPP-net). The set of places in CPP-net consists of three subsets: control places, input places, and constraint places. The elements of the transition set represent the tasks to execute. Based on a Petri net the reachability graph is created. It displays all possible sequences of transitions on the CPP-net which correspond with possible process scenarios [6].

2 Process Planning in the Group Technology Conditions

The way a company executes tasks concerning manufacturing process planning influences its flexibility, efficiency and competitiveness. The CAPP system can be, therefore, a useful tool for facilitating the production effectiveness. Positive effects in the planning process automation can be obtained especially when there are groups/families of technologically-similar parts in a given company. It makes realization of group technology (GT) and the variant approach to process planning possible. Group technology aims at improving production capacity by means of grouping manufactured parts with similar features into families and creating for them one- or multi-machine-tool manufacturing cells. Its implementation in a production company results in organizational improvement, lower production costs and increased efficiency on working stations. It facilitates standardization of manufactured goods and quicker realization of orders, thus improves the relations with clients. Properly used GT system becomes a container in which manufacturing knowledge of a company is stored. However, the optimal configuration and exploitation of manufacturing cells is a complicated mathematical issue [9].

The basic problem in cellular manufacturing is to group the machines into machine cells [11, 13] and the parts into part families [4]. Other important actions are: building a parts classifier [8, 14] developing manufacturing process for composite group technology part [14] developing parametric programs in CNC [3], production scheduling [22], simulation of production capacity in manufacturing cells [19]. A composite part for a given family is a hypothetical part that includes all of the design and manufacturing attributes of the family. A theoretical composite part incorporated all the major features of parts belonging to a family and a machine could be tooled up to produce the composite part, thus providing the set-ups required for each part in the family. In order to identify and group families of technologically-similar parts we can point out three approaches: visual inspection,

production flow analysis, identifying similarities and differences among parts and relating them by means of a coding scheme.

Figure 1 presents the Petri net model divided into functional blocks and information flows. By model blocks we mean coherent subsystems which control the chosen, homogeneous functional tasks of the whole system. The blocks should be formed so that it is possible to put them to the test and verify them separately as autonomous subsystems. The blocks should be characterized by universality meant as usefulness for various types of manufacturing process models. Another desired feature is scalability which enables easy subsystem build-up based on the repeatable fragments of graph topology. In case of a manufacturing process which takes place in a manufacturing cell we can distinguish three categories of functional modules: *Production Tasks*, *Processes* and *Manufacturing Resources Allocation Modules (MRAM)*.

Just like in case of FMS, *Production Tasks* module is the basic model block for the manufacturing process realized in the conditions of a group process. It includes the information about the number of the parts belonging to the family of technologically-similar parts which is necessary to manufacture. Therefore the information is crucial to generate the optimal production schedule in a given production cell.

The *Process* module includes the description of the manufacturing process flows for all the parts manufactured in a given cell. When a limited number of the parts manufactured in the cell the module can consist of autonomous blocks modeling the process flow for only one representative of the family. However, for a substantial number of different but similar parts such approach will cause the overgrowth of a model. It can also make a model more time-consuming if the coding of individual process is done manually. That is why other solutions are being sought here. As far as group technology is concerned we can usually point out several machining steps which fully realize the machining process of all parts in a family. The problem of setting the time for each step remains. Here, we can connect the model with properly configured database. This approach is rather complicated, though. Another, easier solution is the dynamic configuration of the *Process* module so that at the moment it includes the description of only those process, which come out of the current production plan. Such an approach is presented in the following part of the article.

Tools module is used to control the availability of particular tools needed for the realization of manufacturing process. This module controls tool replacement. In a more developed form *Tools* module can control cutting tool durability. Topologically, it consists of the same structures so it can be easily expanded. *Raw Material Availability Control (RMAC)* module controls the availability of raw material and operates the blocks of its replacement. Due to the various technical solutions in that matter *RMAC* should be individually adjusted to a given machine tool. The fixture change within the group technology should be minimal, because the events that occur here are time-consuming. Also for this reason the *Fixtures* module cannot be omitted as the above changes cannot be avoided.

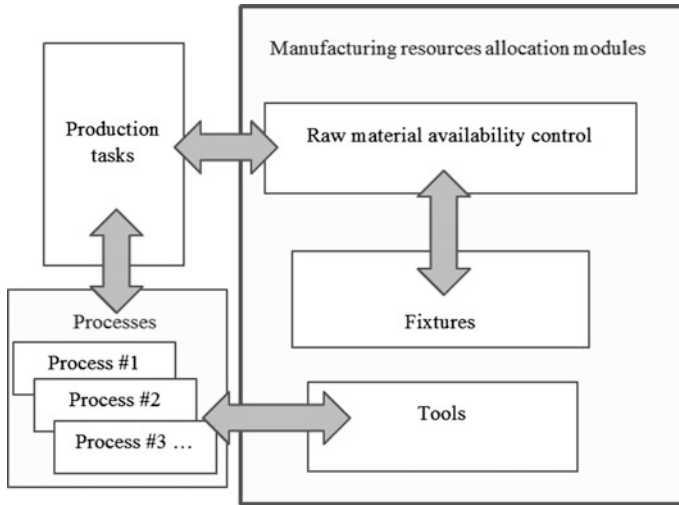


Fig. 1 Information flow diagram of the manufacturing process model

3 Petri Nets Class for Modeling the Performance of Manufacturing Group Process

In order to build the model of manufacturing group process the right class of priority timed Petri nets in the form of ordered nine (1) has been defined:

$$PN' = (P, T, E, K, W, I, S, R, M_0), \tag{1}$$

where:

- P a nonempty, finite set of places,
- T a nonempty, finite, disjointed of P set of transitions,
- $E \subset (P \times T) \cup (T \times P)$ a flow relation,
- K $P \rightarrow N$, a place capacity function,
- W $E \rightarrow N_0$, a weight arc function, N is the set of natural numbers, $N_0 = N \cup 0$,
- I a set of inhibitor arcs,
- S $T \rightarrow N_0$, a function of time [s],
- R $T \rightarrow [0, 1]$, a priority function,
- M_0 $P \rightarrow N_0$, an initial marking

In the graphic interpretation Petri net is a bipartite graph with two disjoint sets of nodes: places (i.e. conditions, positions, resources)—represented by circles; and transitions (events)—represented in a graph by bold bars. Heteronymous nodes are connected by means of directed arcs; their direction shows markers (tokens) flow. It is impossible to connect homonymous nodes. As a graphic element there are also

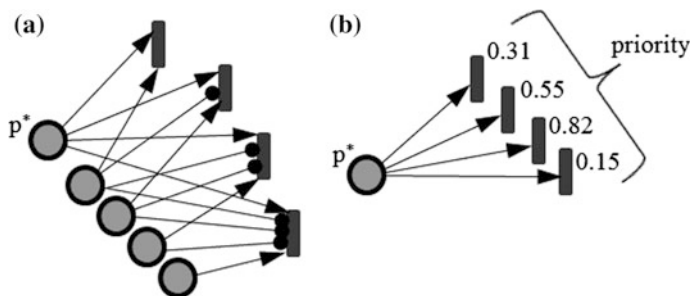


Fig. 2 Conflict resolution

inhibitor arcs presented as the lines with a small circle at the end. The circle indicates the event inhibited by the place connected with it, when the place is active. Thus the inhibitor arcs lead only from place to event. In the discussed class of Petri net only the places with at least one marker are treated as active. Active positions, opposite to inactive ones (in this work) are represented on the graph by dark (green) circle background. Another constraint in event realization is the number of markers in the initial positions of a transition. If the weight of any arc reaching an event is bigger than the current content of a place from which the arc goes, the event cannot start. The remaining elements of Petri net definition are rarely graphically visualized directly on the model. Usually, a designer can access them through the user interface integrated with a graphical window of a simulator. There are many programming tools for designing Petri nets and thus a designer should assess their usefulness before he starts building a model.

The model of condition/event system should settle the conflicts when two or more events compete for the same resources, e.g. the machine tool work time (p^* in Fig. 2). The Petri net class presented in this work can solve conflicts in two ways. In the first one (Fig. 2a) the place set connected with competing events is used. If there are no such positions, they should be set in the model. Then we can establish any firing sequence of transition. When there are many competing events this method leads to a significant extension and makes the model more complicated. Another drawback of this approach are difficulties with automatic sequencing of events.

The other way of conflict resolution (Fig. 2b) uses the priority function described in a chosen subset of competing events. As an example from Fig. 2a, first, we will fire the transition with the priority 0.82 and the one with priority 0.15 as the last. This method is suitable for automatic optimization of competing events sequencing.

4 Petri Net for Group Manufacturing Process

One of the characteristics of a good system modeling tool is its ability to build hierarchical models which represent different steps of modeled system specificity. The constructed model is always an effect of a certain compromise between the precision of a representation of an actual system and the costs of building a model where the work consumption is of the essential importance. Those relations are difficult to capture in the sense of quantity, because the level of building automation plays a considerable role in shaping them. In case of building the model of group manufacturing process it is advisable to ensure the structural resemblance of particular model parts and the possibility of hierarchical extension of the chosen fragments. It facilitates not only the readability of the model but also automation of data coding.

4.1 Production Task Module

The model of *Production Task* module provides an example of an ordered module with potentially easy extension possibility (Fig. 3). This model establishes the order of realization of particular production tasks and controls the moment in which production ends. It has been developed as a timed priority Petri net. The role of inhibitor arcs' is only auxiliary, they are not treated as an element for establishing the task sequence. The example from Fig. 3 has been performed for five production

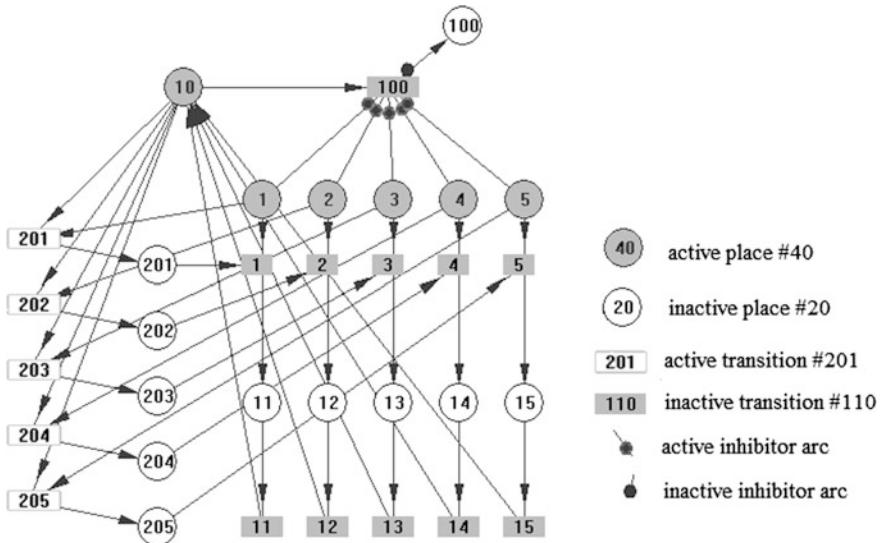


Fig. 3 Model of Production tasks module

tasks represented by the events $t11, t12, \dots, t15$ and each stands for a single manufacturing process. The node which bonds the block into a whole is place $p10$ which demands the continuation of a current task, firing another task, or ending the production. We can observe a typical conflict here, which refers to resources availability, in this case the time of the machining tool work. The solution is provided by the mechanism of the Petri net itself.

Which of the competing options is going to prevail depends most of all on the priority of each task. Those priorities are ascribed to transitions $t201, t202, \dots, t205$. The task with the biggest priority is continued. This mechanism makes the marker to appear only in one position among $p201, p202, \dots, p205$. It is consumed by an appropriate transition $t1, t2, \dots, t5$. At the end, transition $t100$ (the completion of the realization of all production tasks) will be realized. The number of markers in places $p1, p2, \dots, p5$ corresponds with the number of parts to make in tasks 1, 2, ..., 5. The weight of arcs $p1 \rightarrow t1, p2 \rightarrow t2, \dots, p5 \rightarrow t5$ equals 1, only when in one cycle only one part is machined. The weight of arcs $p1 \rightarrow t201, p2 \rightarrow t202, \dots, p5 \rightarrow t205$ equals zero; thus they are only information arcs. The inhibitor arcs set up to $t100$ makes it impossible to end the production before all parts from the particular tasks have been machined according to the plan. Additionally, arc $p10 \rightarrow t100$ guarantees that the machining of the last part has been completed. The remaining elements complement the model so that it meets all formal specifications and to prevent deadlock. The set of positions $p201, p202, \dots, p205$ bears the global meaning for the model. It allows us to choose the steps of process and their correct order as well as availability of fixtures.

The model in Fig. 3 although it is complete does not allow us to perform detailed simulation of production cell work. The model considers only the time of transitions $t11, t12, \dots, t15$, which, in this case, is the general time of the machining cycle including the amount of time during which the raw material has to be delivered to the cutting zone and fixed up. The model does not take the additional time needed for tool replacement, a tip of a bar removal, the change of fixtures, etc. into consideration. Considering the ease of building such a model it should be estimated as relatively useful. It does not require optimization procedure as the change of sequence in which particular tasks are to be realized does not influence the amount of time during which all tasks are completed.

4.2 Raw Material Availability Control (RMAC)

Figure 4 presents an RMAC module for the model manufacturing process along with the links to *Production tasks* module. Its structural form may differ depending on the fixture system of the workpiece change. The model presented here refers to the process on the CNC lathe where the raw material in the form of cylindrical rods or pipes is delivered manually, from the front of the spindle. It allows us to make several parts from a rod 1 m long in one automatic cycle depending on the length of a part. Thus the model should control, apart from the number of parts to make, the

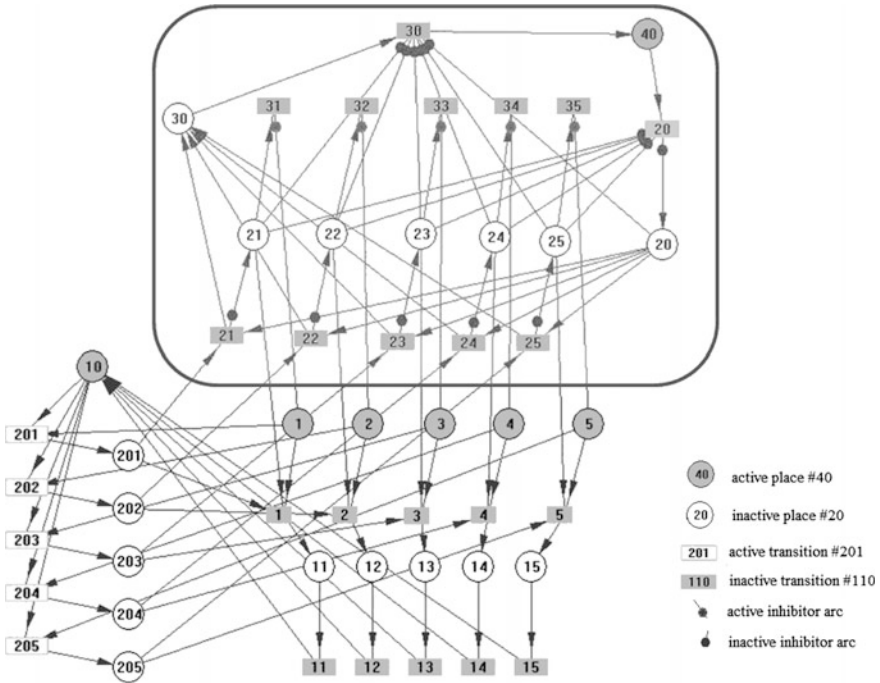


Fig. 4 Model of *RMAC* modul

number of parts that can be made of currently fixed workpiece, the tips of the rod removal and fixing the next piece of raw material.

The basic element of the *RMAC* module structure is the set $p_{21}, p_{22}, \dots, p_{25}$, which stores the number of parts of a given type that can be made out of the raw material currently fixed. Elements of that set are connected with placing the material in the cutting zone t_1, t_2, \dots, t_5 and the transition set $t_{31}, t_{32}, \dots, t_{35}$ which are responsible for removing the unused rod endings from the fixture. Path $p_{30} \rightarrow t_{30} \rightarrow p_{40} \rightarrow t_{20} \rightarrow p_{20}$ has a global character. It controls the raw material replacement in case when the type of machined parts changes. The weight of arcs $p_{21} \rightarrow t_{21}, t_{22} \rightarrow t_{22}, \dots, p_{25} \rightarrow t_{25}$, equals the number of parts that can be made out of one piece of raw material.

The initial state of *RMAC* module is presented on Fig. 4. The only active place in the model is p_{40} (the demand for the next piece of raw material), which fires the transition t_{20} (permission to load). The details of *RMAC* module can analyze the successive steps of model processing (Figs. 5, 6, 7, 8, 9 and 10).

Step 1: Firing of the transition t_{20} activates p_{20} (permission to load a rod), which creates the conditions for firing one of the transitions $t_{21}, t_{22}, \dots, t_{25}$ (loading the raw material of the right type). These are mutually exclusive events. Only one raw material can be loaded, in this case #4, as *Production*

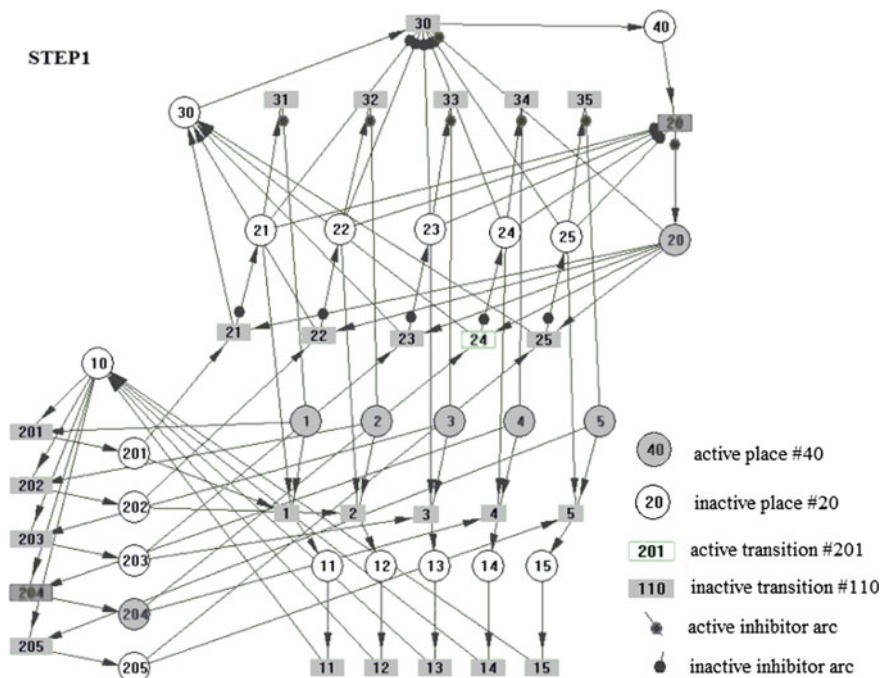


Fig. 5 State of model after step 1

Tasks module simultaneously activated position p204. Through the arcs $p201 \rightarrow t21$, $p202 \rightarrow t22$, ..., $p205 \rightarrow t25$, flows only the information without the markers, because the weight of those arcs equals 0.

- Step 2: The decision to choose transition $t24$ was made on the basis of event priority ranking $t201, t202, \dots, t205$. The firing of transition $t24$ delivers $p24$ the number of markers that equals the number of parts #4 that can be made out of one piece of raw material. That is why the weight of arc $t24 \rightarrow p24$ equals this number. Activating position $p24$ enables the firing of transition $t4$, consequently, the machining of part #4. At the same time, position $p24$ inhibits the firing of transition $t30$ (removal of the rod tip), by means of an inhibitor arc.
- Step 3: After unchucking the material ($t4$) it is possible to machine part #4 ($t14$). The state of *RMAC* module do not undergo any changes at that time.
- Step 4: This step illustrates the situation after machining all parts of #4, which corresponds to removing all the markers from position $p4$. At the same time the transition $t4$ removed the markers from position $p24$. Along with the firing of the transition $t4$ for the last time, the transition (time = 0) $t34$ is fired and it removes the remaining markers from position $p24$. Then the inhibitor arc from $p24$ to $t30$ will not block the transition $t30$, so the path to load the next piece of raw material will initiate.

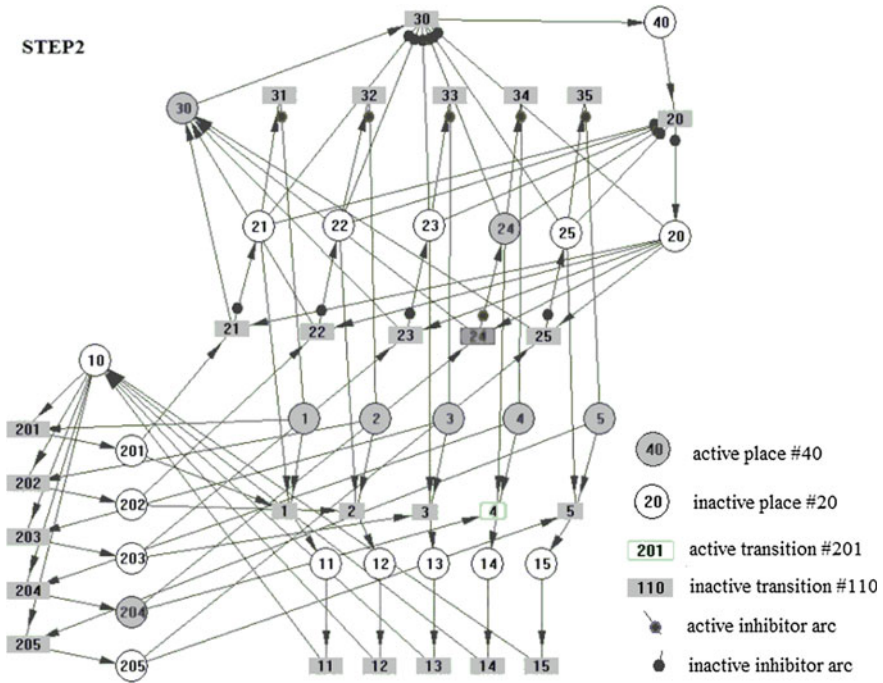


Fig. 6 State of model after step 2

Step 5: When t30 is completed the net return to its initial state (Fig. 4), but with one little exception: there are no more markers in position p4. As a result, another type of part to be machined with the highest priority (#3) is chosen.

Final Step: The final state of *RMAC* module is the same as the initial state. However, because there are no more markers in positions p1, p2, ..., p5, there is no more processing and the net becomes blocked.

4.3 The Module for the Fixture Element Selection

Another module closely related to *RMAC* is *Fixtures* module. Its form depends heavily on the type of machining tool and the way a workpiece is loaded and unloaded. The presented example of a model refers to the CNC lathe with automatic, hydraulic, self-centering chuck (Fig. 11) that is different for different shapes of raw material. During the production process such activities as: changing the chuck (approx. 2 h), changing the jaws (approx. 20 min.) or the change of jaws position (approx. 10 min.). It amounts to a long time compared to the time of one machining cycle. Thus the schedule of machining tool work should minimize the

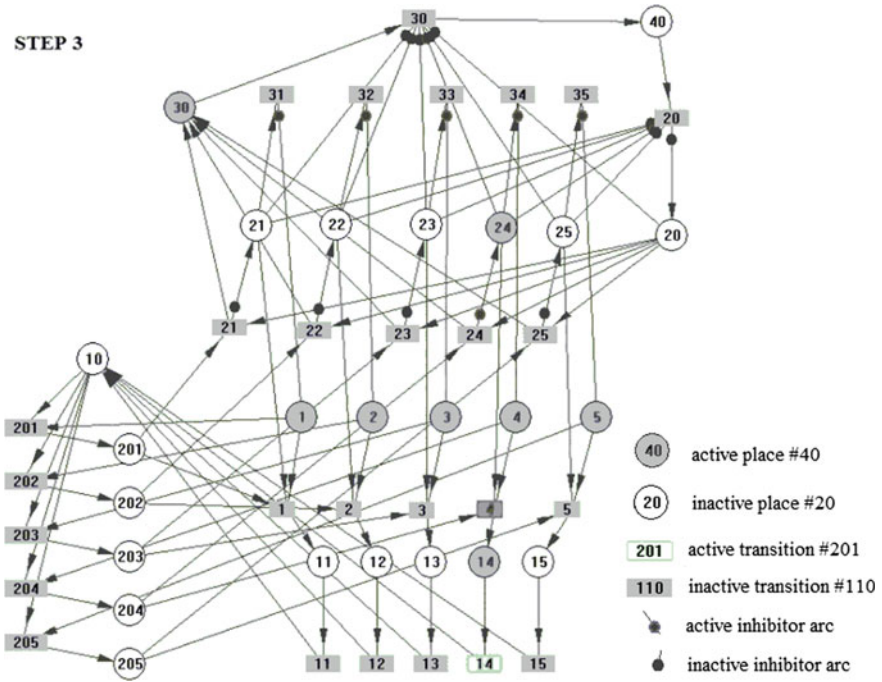


Fig. 7 State of model after step 3

number of those actions. *Fixtures* module and its connections to *Production Tasks* and *RMAC* is presented in Fig. 12.

Figure 12 shows *Fixtures* model for only one type of lathe chuck. It has been assumed that each of the three sets of jaws can be placed in three different positions which secures the necessary range of fixing. As can be seen in Fig. 12, despite simplifications, the *Fixtures* module is the most extended and complex module in the whole model. The positive thing is that it consists of blocks, which are structurally the same, joint for the chuck (when we assume its change), jaws set and the position of jaws in the chuck. Such a block has only two places: input (p215) and output (p115). An active input place means that currently chosen production task wants to establish a given position of jaws in the chuck. An active output place enables the loading of the right raw material. Four transitions present in the block attend to those places. Transition t235 consumes the tokens from position p215 when p115 is active. Transition t215 activates place p115 on condition that none of inhibitor arcs reaching t215 is active that is no other position of jaws set is active. Firing of transition t215 requires the marker in the position p120 that is the permission to change the position of the jaws. Transition t245 activates the permission to change the position of jaws and then t225 consumes the marker in the position p115. The weight of arcs $p215 \rightarrow t245$ and $p115 \rightarrow t235$ equals zero. As seen in Fig. 12, positions p110 and p120 are polar nodes for the whole group of

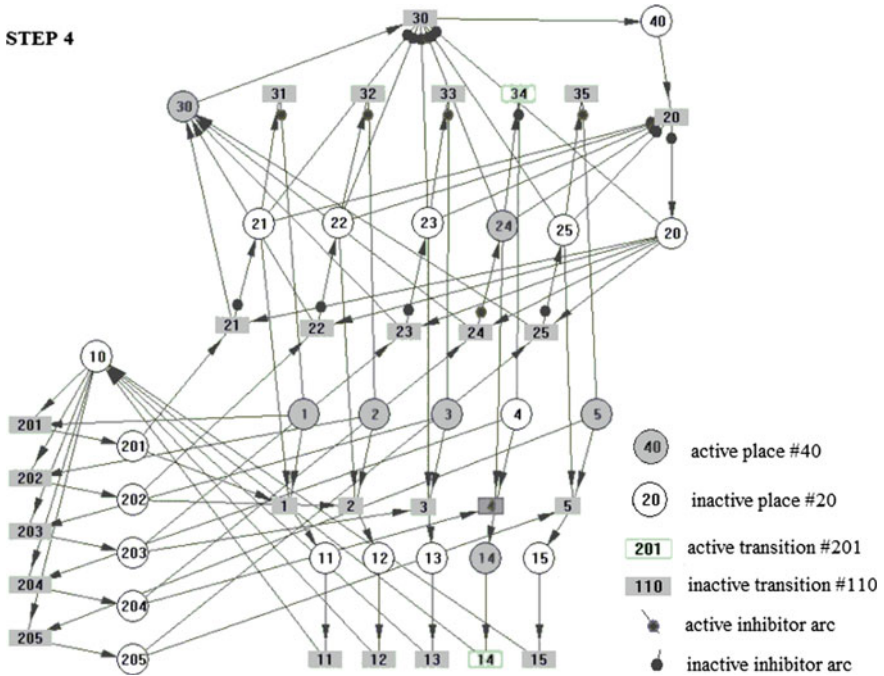


Fig. 8 State of model after step 4

homogeneous blocks. They exclude simultaneous activity of competing facts. The events of changing the position of jaws in the chuck overlaps the replacement of jaw set in the chuck and it is taken into account in *RMAC* module.

4.4 Manufacturing Processes Module

Process module allows us to record the sequences of steps of process for all production tasks. As their number can be quite big (even a few thousand) they should be limited only to those executed in relatively short periods of time. Therefore completing this module should be automated and integrated with databases or CAPP databases (Table 1) or Production Flow Analysis.

The principal condition for planning the correct manufacturing process is the existence of Feature Precedence Network (FPN). FPN is defined as directed graph representing precedence's resulting from the constraints imposed by the features existing in a given part. It is worth noting that FPN represents only the constraints resulting from structural interactions of particular features and it does not consider such aspects of the manufacturing process as machining of a few features in the same position, with the same tools, etc. Petri nets do not have such constraints.

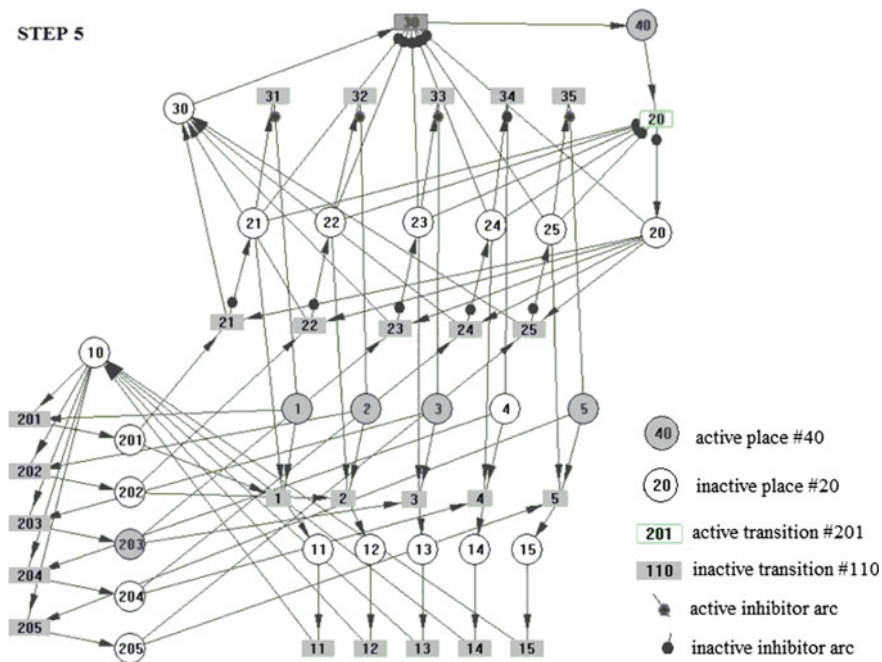


Fig. 9 State of model after step 5

In case of the manufacturing process realized on common CNC lathe the net representing the process is a simple serial chain of the steps of manufacturing processes and the current state of machined part, alternately. However, when we take a lathe with the bigger number of machine units and/or spindles into account the process undergoes quick expansion. The pairs of parallel steps of process which are realized at the same time. In such case, we should consider building alternative manufacturing process runs as establishing the constant sequence of actions during the realization of consecutive cycles limits the manufacturing ability of such lathe. Petri nets can model all those conditions.

Figure 13 exemplifies the model of *Processes* module for 5 production tasks. We should bear in mind that the variable number of production tasks brings about the necessity of adjusting the *Processes* module and *Production Tasks* module as well. The former has a simple, clear structure, because it consists of serial chains of places (the state of process) and single steps of process. It facilitates developing an integrator which automatically generates the data to the model. To make such a system user-friendly it should possess the right user interface. The marker from position p10 is the confirmation of the machining cycle completion and at the same time it triggers the next cycle by the events sequence t201, t202, ..., t205.

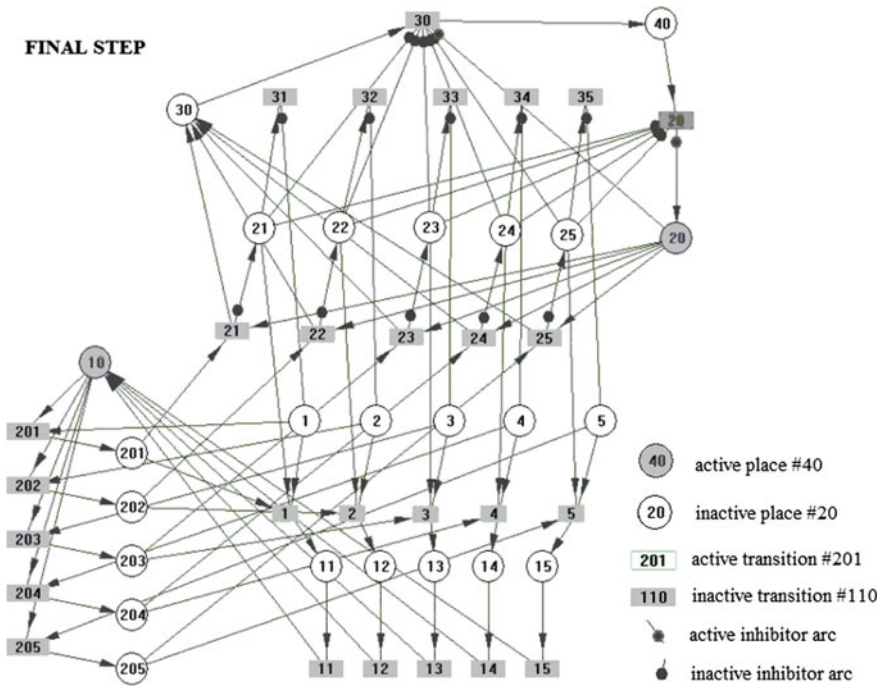


Fig. 10 State of model after final step

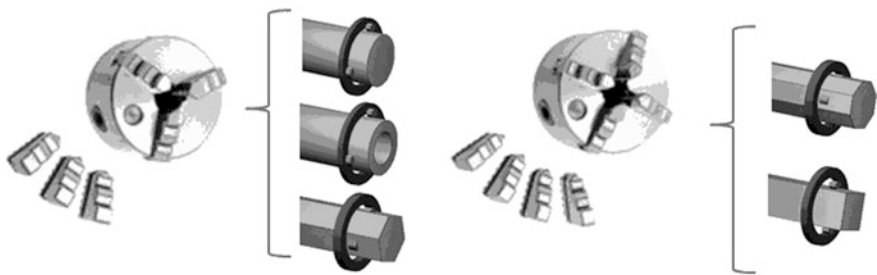


Fig. 11 Self-centering lathe chucks

4.5 Tools Module

The last of the modules presented controls of the right tool for a given step of machining process is available. If not, the demand for such a tool is activated. This module is closely related with the *Processes* module. Before a given step of the process begins, previously fired transition demands access to a tool needed in this

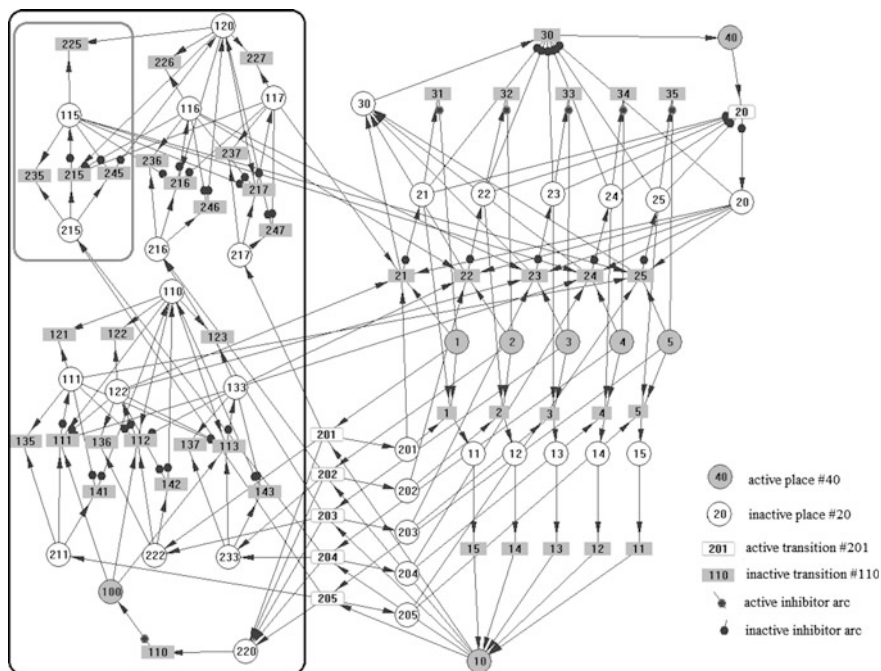











Fig. 12 Module fixtures

Table 1 Set of workpieces and machining steps

Machining steps									
Turning	*	*	*	*	*	*	*	*	*
Drilling				*	*	*	*	*	*
Boring	*	*		*	*	*			*
Grooving						*			
Partitioning	*	*	*	*	*	*	*	*	*

step. When we have another tool in the working position, the procedure of changing the tool is initiated.

Structurally, Tools module consists of blocks (Fig. 14 exemplifies 5 tools have been modeled) of the same structure identical to the blocks of the access to the chuck elements. In each block, there are two input places (p4*0) and output places (p4*1) through which the block exchanges information with the Processes module. The node connecting all the blocks together is p400. The number of tools in case of group technology is usually limited as the universal tools are in use and there is a tendency to change them as rarely as possible, so the size of Tools module is not

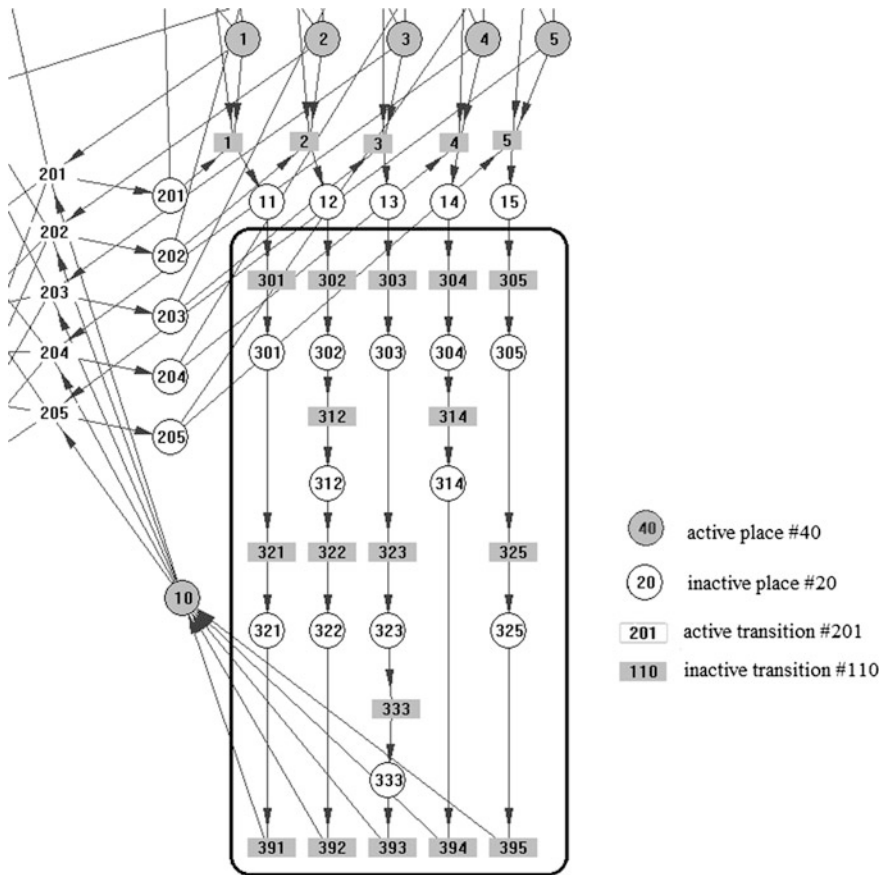


Fig. 13 Processes module

significant. Tools module is constant unless there is a necessity of fill in the tool warehouse. Between the consecutively realized production programs the connection between *Tools* and *Processes* should be configured.

In case of tool management models, usually, the cutting tool control function is modeled as well. There are two approaches: the number of cycles one cutting tool is able to perform or the control of summary time of cutting tool work. For the Petri net class presented here only the former approach is possible to model. Structurally, cutting tool durability block (Fig. 13) can be modeled in a very simple form by use of a single place and a single transition. Positions p501, p502, ..., p505 store the number of cycles that one cutting tool is able to perform until it wears and has to be replaced. Transitions t501, t502, ..., t505 store the action of changing the cutting tool when the demand for this tool occurs and the number of cycles it can perform equals zero. The weight of arcs p410 → t501, ..., equals zero. The weight of arcs t501 → p501 is equal to the predicted durability of a new cutting tool expressed in

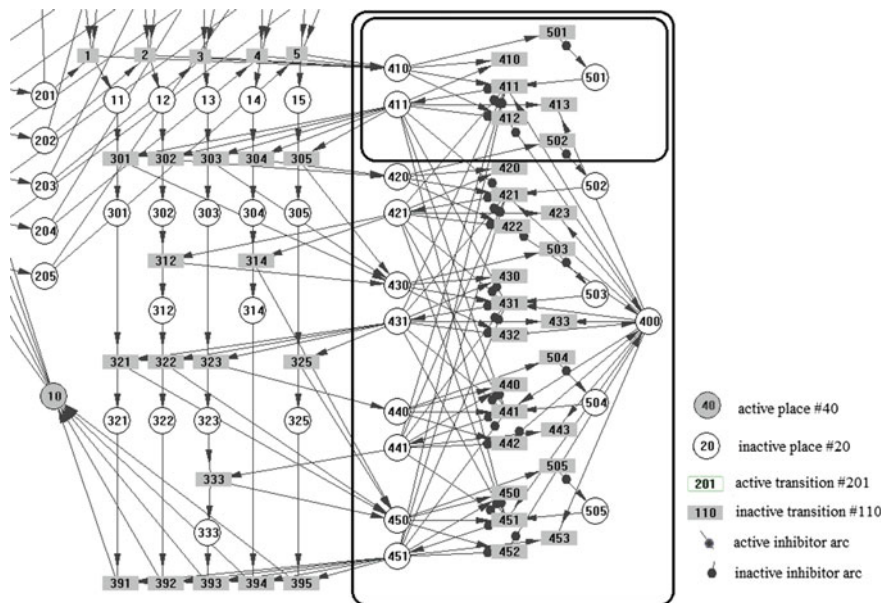


Fig. 14 Tools module

cycles. Acceptance of this type of solution is sort of a compromise, as a given tool machines various parts to a various degree. Thus the actual wear of cutting tool for each machined part differs. The method applied assumes a certain average wear per one machining cycle. This approach is partially justified when we assume that we deal with a manufacturing cell which machines part of a similar size and shape.

5 Optimization of Production Program Realization in Automatic Manufacturing Cell

Having a time model of a process expressed in the language of priority Petri nets we can use it as a platform for generating the flow schedule of consecutive production tasks through the manufacturing cell. Optimization of these actions is possible when the priorities of a chosen group of events are established correctly. In the presented example it was the group p201, p202, ..., p205 (Fig. 3), of events which indicate the type of consecutive manufactured parts. The successful approach when optimizing this type of problems is to use metaheuristic in the form of properly configured genetic algorithm. Genetic algorithms (GA) are computing algorithms constructed in analogy with the process of evolution. Now, this method could be used to solve a variety of different problems, such as training of neural nets,

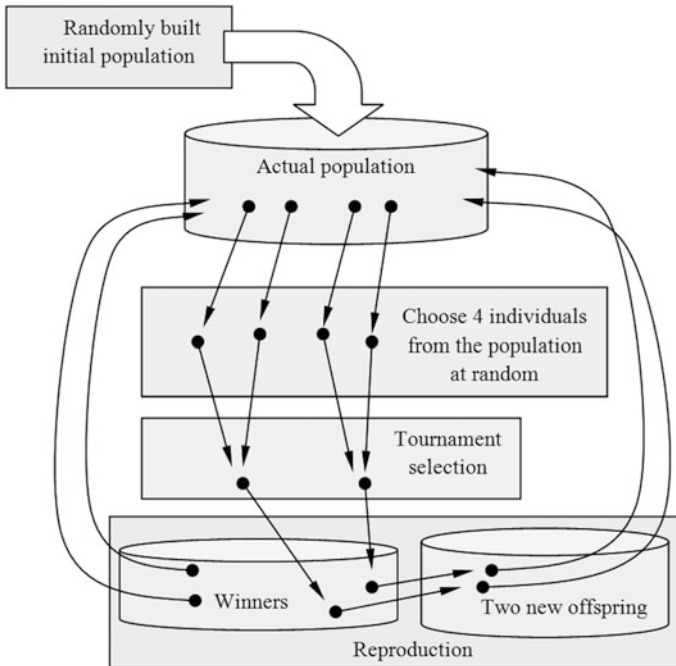


Fig. 15 Flow chart for genetic programming

extraction of image feature extraction, image feature recognition, process route sequencing and also job scheduling.

The version of GA used here is characterized by elitism. The winners of the competition participate in a reproduction and along with their descendants move to the next generation (Fig. 15). All chromosomes of the first generation are randomly generated (Fig. 16).

where:

- P_1 the first parent index,
- P_2 the second parent index,
- N new offspring index,
- Random randomly selected number in the range of $[0, 1]$,
- V_i value of i th gene in the range of $[0, 1]$,
- n number of genes in chromosomes,
- MP mutation probability,
- L probability that the first parent will pass the i th gene

A single chromosome represents exactly one solution. The form of chromosome in this method can be very natural because it is an ordered set of priorities of particular production tasks. Solution of the problem corresponds to the priority function R . The number of genes in a chromosome in this case equals the number of activated production tasks to fulfill. Each gene represents priority of a single transition (Fig. 17). The priorities take values from the range of $[0, 1]$. Therefore the

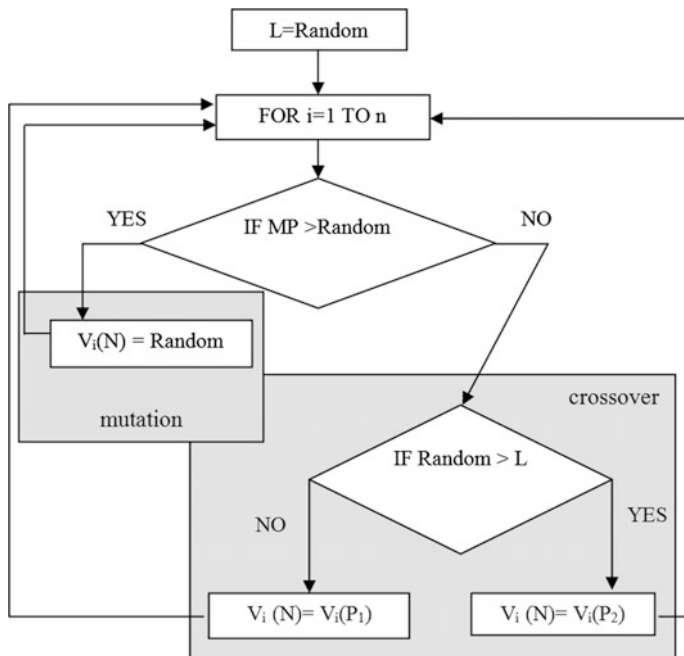


Fig. 16 Flow chart for reproduction

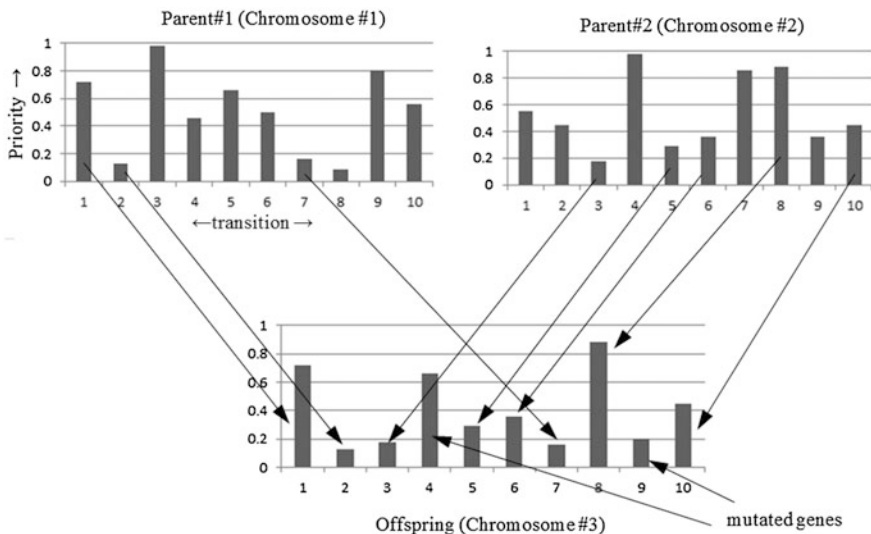


Fig. 17 Chromosomes reproduction scheme in genetic algorithm

Petri net model, if used properly, makes it possible to optimize the work of the manufacturing cell, also when the production program changes dynamically.

6 Summary

The mechanisms proposed in the Petri network enables, apart from convenient modeling of conditions and sequences of all events comprising the manufacturing process, controlling the availability of production resources along with analyzing the time series. The priority function allows easy application of metaheuristics optimizing the work of a manufacturing cell. The Petri net in the above configuration has some constraints which make the precise control over some resources in a complex manufacturing system impossible; e.g. the precise control over the tool work time or the use of raw material in the manufacturing of different parts. The class can be extended with three, additional elements: the variables set, the set of positions containing sentences of a predicate logic with the use of variables and the set of events realizing calculations on defined variables. There is the principal that the real condition is generated by the marker in the position the condition is connected with. Due to the introduction of those three elements the possibility to model the process increases definitely. The Petri nets in so extended form can be compared to the conclusion mechanism of expert system. Unfortunately, the time consumption of building such a model rises and so does the cost. Therefore it should be assumed that the solution presented in this work is a rational compromise between potential possibilities and the time consumption of model building.

References

1. Banaszak, Z. (ed.): *Modelling and Control of FMS, Petri Nets Approach*. Wrocław Technical University Press (1991)
2. Cecile, J.A., Srihari, K., Emerson, C.R.: A review of Petri net application in process planning. *J. Adv. Manuf. Technol.* **7**, 168–177 (1999)
3. Djassemi, M.: An efficient CNC programming approach based on group technology. *J. Manuf. Syst.* **19**(3), 213–217 (2000)
4. Galan, R., Racero, J., Eguia, I., Garcia, J.M.: A systematic approach for product families formation in reconfigurable manufacturing systems. *Rob Comput. Integr. Manuf.* **23**, 489–502 (2007)
5. Kiritsis, D., Porchet, M.: A generic Petri net model for dynamic process planning and sequence optimization. *Adv. Eng. Softw.* **25**, 61–71 (1996)
6. Kiritsis, D., Xirouchakis, P., Gunther, C.: *Petri Net Representation for the Process Specification Language—Part 1: Manufacturing Process Planning*. International Workshop on Intelligent Manufacturing Systems IMS-EUROPE, Lausanne (1998)
7. Kumar, R.R., Singh, A., Tiwari, M.K.: A fuzzy based algorithm to solve the machine-loading problems of a FMS and its neuro fuzzy Petri net model. *Int. J. Adv. Manuf. Technol.* **23**, 318–341 (2004)

8. Kuric, I.: Dynamic classification in group technology. In: 7th International Multidisciplinary Conference, Baia Mare, Romania. ISSN-1224-3264 (2007)
9. Kusiak, A.: The generalized group technology concept. *Int. J. Prod Res* **25**(4), 561–569 (1987)
10. Lee, K.H., Jung, M.Y.: Flexible process sequencing using petri net theory. *Comput. Ind. Eng.* **28**(2), 279–290 (1995)
11. Nouri, H., Tang, S.H., Hang Tuah, B.T., Anuar, M.K.: BASE: a bacteria foraging algorithm for cell formation with sequence data. *J. Manuf. Syst.* **29**, 102–110 (2010)
12. Pawlewski, P. (ed.): *Petri nets – manufacturing end computer science*. Novi Sad, InTech (2012)
13. Ribeiro, J.F.F.: Manufacturing cells formation based on graph coloring. *J. Serv. Sci. Manage.* **3**, 494–500 (2010)
14. Roy, D., Sarma, U.M.B.S., Tiwari, M.K., Mukhopadhyay, S.K.: A part classification system using a shape-analysis based heuristic approach. *Int. J. Adv. Manuf. Technol.* **21**, 355–364 (2003)
15. Roy, N. Komma, V.R.: Cellular manufacturing through composite part formation: a genetic algorithm approach. In: *International Conference on Industrial Engineering and Operations Management*, Bali (2014)
16. Rudas, I.J., Horvat, L.: Modeling of manufacturing processes using a Petri-net representation. *Eng. Appl. Artif. Intell.* **10**(3), 243–255 (1997)
17. Srihari, K., Emerson, C.R.: Petri nets in dynamic process planning. In: *12th Annual Conference on Computers and Industrial Engineering*, pp. 447–451. Orlando (1990)
18. Stryczek, R.: Petri net-based knowledge acquisition framework for CAPP. *Adv. Manuf. Sci. Technol.* **32**(2), 21–38 (2008)
19. Stryczek, R.: A meta-heuristics for manufacturing systems optimization. *Adv. Manuf. Sci. Technol.* **33**(2), 23–32 (2009)
20. Tuncel, G., Bayhan, G.M.: Applications of Petri nets in production scheduling. *Int. J. Adv. Manuf. Technol.* **34**, 762–773 (2007)
21. Uzam, M.: The use of the Petri net reduction approach for an optimal deadlock prevention policy for FMS. *Int. J. Adv. Manuf. Technol.* **23**, 204–219 (2004)
22. Wang, J.B., Lin, L., Shan, F.: Single-machine group scheduling problems with deteriorating jobs. *Int. J. Adv. Manuf. Technol.* **38**, 808–812 (2008)
23. Zhou, M.C., Venkatesh, M.: Modeling, simulation and control of flexible manufacturing systems. In: *Intelligent Control and Intelligent Automation*, vol. 6, pp. 43–44. World Scientific Publishing Co., Singapore (1999)

Graph Theory in Product Development Planning

I. Kutschenreiter-Praszkiewicz

Abstract Development of process planning methods is an important research area. Product development planning takes into consideration lead-times of tasks and activities, bill of materials, quality plans, and risk analysis. Production process planning can be supported by graph theory. Methods such as CPM, PERT, GERT are useful in product development planning. In this article, a methodology of risk identification and assessment is combined with the GERT method.

Keywords Product development planning · GERT · Risk assessment

1 Introduction

Product development has to meet tight budgets and deadlines [1]. The aim of this article is to join planning methods based on graph theory with the risk management method.

Uncertainties and risk connected with product development projects comes from different sources and they have to be modelled respectively [2].

Deciding about product development, several criteria should be taken into consideration simultaneously. The most important of them are: financial criteria (project should consume limited costs), time criteria (project should be finished in given time) and results criteria (project should be focused on particular results).

In order to be able to integrate multi-dimensional criteria of project realization and evaluation, it is necessary to aggregate evaluation criteria and combine them with a stochastic method of project planning.

I. Kutschenreiter-Praszkiewicz (✉)
University of Bielsko-Biała, Bielsko-Biała, Poland
e-mail: ipraszkiewicz@ath.bielsko.pl

Stochastic planning methods help to create schedules that take into consideration all possible solutions before the final schedule is identified. The most common stochastic planning methods are PERT and GERT [3].

GERT is a graph based project planning method that can handle the successive probability of possible project outputs.

2 GERT

Graphical Evaluation and Review Technique, commonly known as GERT, is a network analysis technique used in project management that allows probabilistic treatment of both network logic and estimation of activity duration [4–7]. The technique was first described in 1966 by Pritsker [8]. GERT allows loops between tasks and application of three types of nodes in the graph:

- node “and” marked in the graph as a circle indicates that all tasks have to be done,
- node “or” marked in the graph as a 45° rotate square indicates that at least one of the previous tasks has to be done,
- node “either” marked in the graph as a 45° rotated square with the line perpendicular to the square diagonal means that exactly one from the previous tasks has to be done.

A GERT graph is built with the one start node and some end nodes, which means different possibilities of project ending are possible.

In a GERT graph, edges indicate tasks for which project resources, such as time or costs and probability are allocated.

To create a GERT graph, it is necessary to follow the steps below:

- first step: transform linguistic discretion of the project into a stochastic graph. Indicate project tasks and relations between them,
- second step: determine data regarding project tasks. Fix tasks duration or costs and their probability,
- third step: determine substitute transmutation—which gives information regarding probability of project success and expecting total project time or costs. Graph has to be reduced according to principles given in Table 1,
- fourth step: interpret the results.

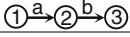

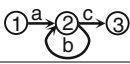
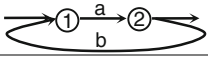
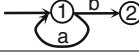
The expected time of a reduced graph can be calculated according to formula (1) [3]:

$$\mu_{1(1,n_k)} = \frac{\sigma}{\sigma_s} W_{1,n_k}(s) \Big|_{s=0} \quad (1)$$

where:

$W_{1,n_k}(s)$ graph transmittance between node 1 and n_k .

Table 1 Graph algebra

Sub graph	Sub graph probability calculation
	$p_e = p_a p_b$
	$p_e = p_a + p_b$
	$p_e = \frac{p_a \cdot p_c}{1 - p_b}$
	$p_e = \frac{p_a}{1 - p_a p_b}$
	$p_e = \frac{p_b}{1 - p_a}$

The presented approach used is a moment generating function that helps finding the mean and the variance of a random variable. Variance can be calculated according to formula (2) [3]:

$$\sigma_{1,n_k}^2 = \text{Var}(X) = \mu_{2(1,n_k)} - [\mu_{1(1,n_k)}]^2 \tag{2}$$

where:

$$\mu_{2(1,n_k)} = \frac{\sigma_s^2}{\sigma_s^2} W_{1,n_k}(s)|_{s=0} \tag{3}$$

Probability related to tasks can be assessed with the use of a risk management method.

3 Risk in Product Development

Risk can be described as the chance of an event happening which may have an impact on project objectives [9].

Risk can arise from different sources and have different consequences. The source of risk may be outside of the organization (external risk), as well as inside organizations (internal risk). External risk can be caused by natural environment (earthquake, flood, etc.), sociocultural factors (strike, disturbances, etc.), technological factors (new materials, new technology, etc.), economic factors (financial crisis, business partners instability, ...), political/legal factors (e.g. new regulations can cause opportunity or threat for an enterprise).

Risk could cause additional project cost, extended project time, influence project results quality, etc.

Additional project activity needed in case of a risk event happening can be managed with the use of a chosen risk management standard.

The procedure of risk management includes the following steps [10]:

- establishing risk context—identifying field of risk analysis (external context, internal context, risk criteria,...),
- risk identification—defining what can happen and why, when, where, how,
- risk analysis—estimating level of risk, determining the probability and consequences of risk future occurrence,
- risk evaluation—risk classification into given groups (e.g. low, medium, and high). For that purpose a risk matrix could be created,
- risk treatment (risk can be: accepted, transferred or reduced)

Risk management processes can be combined with the GERT method (Fig. 1). Results of risk analysis which are possible disturbances in project realization can be integrated with a project planning method. Among different project planning methods, like e.g. Critical Path Method (CPM), Program Evaluation and Review Technique (PERT) or the Gant chart, only GERT allows for back coupling modelling, taking into account their probability and consequences. GERT also allows for variants modelling of a project task. CPM is a deterministic planning method

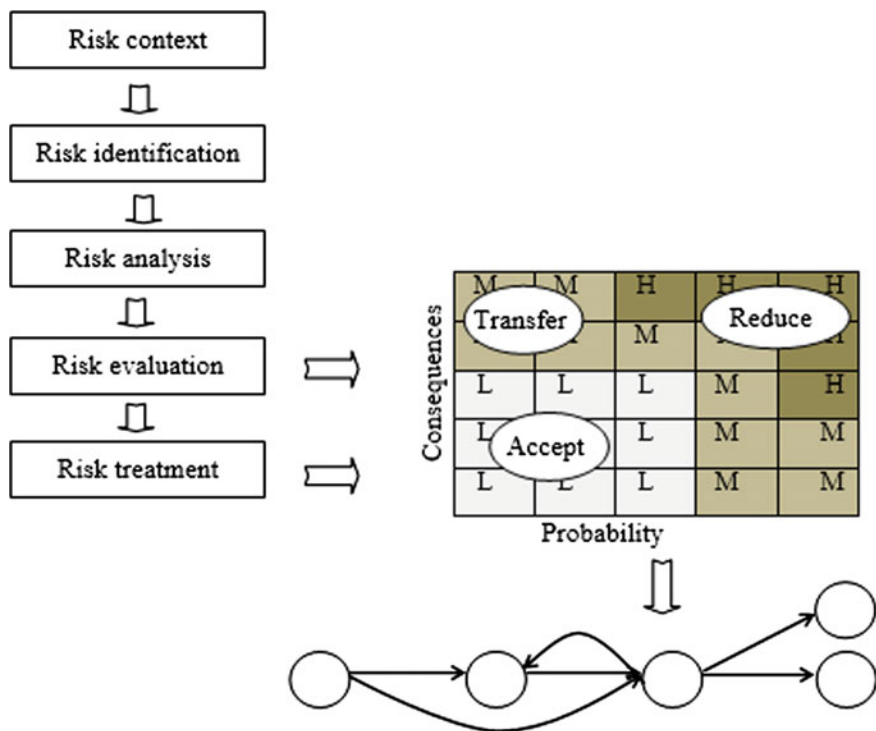


Fig. 1 Risk in product development planning

that needs precise data related to a project task. PERT takes into consideration uncertainty related to project task duration but it is not possible to model different variants of a project task.

A project focused on product development can be modelled with the use of a GERT graph, which includes main project tasks as well as risk that can occur during project realization. Project tasks could be performed in concurrent ways, but some of the project tasks could be completed with the use of different methods or tools. In the beginning of a project, the objectives are defined but the way to achieve them can be different, so the alternatives should be modeled in an applied planning method. Also, the residual risk and activity needed to reduce medium and high risk should be modelled using a product development planning method.

4 Risk Management and the GERT Method: A Joint Example

The example is focused on project planning with joint risk management and the GERT method. Any enterprise has to distinguish tasks in order to develop product. Each task must end with given results. The following tasks were specified in the example of a toothed gear production process planning of a customized product:

- brief foredesign,
- engineering documentation,
- production documentation,
- processing,
- tests.

In the presented approach, risks were identified and assessed. The following risks were identified:

- (a) foredesign failure,
- (b) engineering documentation failure,
- (c) production documentation failure,
- (d) product failure,
- (e) project abandonment.

Failure probability was assessed as 0.01. Risk consequences in the case of forced foredesign failure (a) are brief foredesign repeating and correction which takes approximately 2 h. Engineering documentation failure (b) brings about 12 h, product documentation failure (c) brings about 2 h, product failure (d) brings about 8 h. The identified risks were presented in a risk matrix (Fig. 2).

The identified risks were accepted and the following tasks were added to the list of project tasks:

Consequences	≤40	M	M	H	H	H
	≤16;40>	M	M	M	M	H
	≤8;16>	L (b) (d)	L	L	M	H
	≤1;8>	L (a) (c)	L	L	M	M
	<1	L (e)	L	L	M	M
		<0,05	≤0,05;0,1>	≤0,1;0,2>	≤0,2;0,5>	≤0,5
Probability						

Fig. 2 A risk matrix example

- foredesign correction,
- engineering documentation correction,
- production documentation correction,
- product correction.

Each task had a determined probability of occurrence and moment-generated functions of task times. For each task transmittance was determined (Table 2). Numerical values were based on a set of examples.

Table 2 Product development tasks

Project task	Task description	Task dependency	p_i	M_i	$W_i = p_i * M_i$
A	Brief fore design	Start	1	e^{12s}	e^{12s}
B	Engineering documentation	After A	1	e^{20s}	e^{20s}
C	Production documentation	After B	0.99	e^{10s}	$0.99e^{10s}$
D	Processing	After C	0.9	$e^{74.46s}$	$0.9e^{74.46s}$
E	Tests	After D end task	1	e^{8s}	e^{8s}
F	Fore design correction	Parallel A, opposite direction	0.01	e^{2s}	$0.01e^{2s}$
G	Engineering documentation correction	Lap after B	0.01	e^{12s}	$0.01e^{12s}$
H	Production documentation correction	Lap after C	0.01	e^{2s}	$0.01e^{2s}$
I	Product correction	Lap after D	0.01	e^{8s}	$0.01e^{8s}$
J	Project abandonment	End after A	0.01	e^s	$0.01e^s$
K	Project abandonment	End after B	0.01	e^s	$0.01e^s$
L	Project abandonment	End after C	0.01	e^s	$0.01e^s$

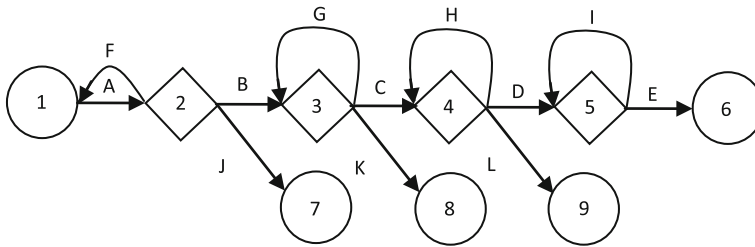


Fig. 3 GERT graph of product development project

The GERT graph of the product development tasks described in Table 2 was presented in Fig. 3. Edges in the presented graph indicate project tasks, which were denoted by capital letters. Letters from A to E indicate main project tasks and letters from F to L indicate tasks which come from a risk management method.

To calculate probability of project success it is necessary to determine substitute transmittance. Equation (4) presents the dependency between node 1 and 6 related to the graph presented in Fig. 3.

$$W_{1-6} = \frac{W_A \cdot W_B \cdot W_C \cdot W_D \cdot W_E}{(1 - W_A \cdot W_F) \cdot (1 - W_G) \cdot (1 - W_H) \cdot (1 - W_I)} \tag{4}$$

$$W_i = p_i \cdot M_i \tag{5}$$

where:

W_i transmittance of i task,

p_i probability of i task,

M_i moment—generated function of i task time

Based on the data presented in Table 2 probability of project success was determined:

$$W_{1-6} = \frac{e^{12s} \cdot e^{20s} \cdot 0.99 \cdot e^{10s} \cdot 0.9 \cdot e^{74.46s} \cdot e^{8s}}{(1 - 0.01 \cdot e^{2s} \cdot e^{12s}) \cdot (1 - 0.01 \cdot e^{12s}) \cdot (1 - 0.01 \cdot e^{2s}) \cdot (1 - 0.01 \cdot e^{8s})}$$

$$W_{1-6} = 0.928.$$

Expected total time of the project was calculated according to formula (6) where the transmittance is differentiating at zero:

$$\mu_{1(1,6)} = \frac{\sigma}{\sigma_s} W_{1,6}(s)|_{s=0} \tag{6}$$

$$\mu_{1(1,6)} = 116.281h.$$

Total project time variance was calculated according to formulas (7) and (8) with the use of a second derivative:

$$\sigma_{(1,6)}^2 = \mu_{2(1,6)} - [\mu_{1(1,6)}]^2 \quad (7)$$

where:

$$\mu_{2(1,6)} = \frac{\sigma^2}{\sigma_s^2} W_{1,6}(s)|_{s=0} \quad (8)$$

$$\sigma_{(1,6)} = \sqrt{\mu_{2(1,6)} - [\mu_{1(1,6)}]^2} = 32.542.$$

5 Conclusions

The presented approach can be used in an enterprise for product development planning. Those projects are characterized by lack of knowledge regarding project tasks. It is possible that a project will be completed before achieving its project goal. Commonly used project planning methods, such as CPM and PERT, are not sufficient.

Enterprises implement risk management procedures, therefore the GERT method which helps integration of product development planning with risk management is promising.

Lack of knowledge regarding project tasks can be modelled with the use of probability; also risk is modelled with the use of its consequences and probability.

In project development planning and risk analysis, probability can be assessed by frequency analysis or set up by experts. The weak side of the proposed approach can be lack of statistics which help in probability assessment or lack of experts who will be able to assess probability.

References

1. Eckert, C., Clarkson, J.: Planning development processes for complex products. *Res. Eng. Design* **21**, 153–171 (2010)
2. Werners, B., Weber, R.: Decision and planning in research and development. In: Zimmermann, H. (ed.) *Practical Applications of Fuzzy Technologies*. Springer, US (1999)
3. Kukuła, K.: *Badania operacyjne w przykładach i zadaniach*. PWN, Warszawa (2004)
4. Kurihara, K., Nishiuchi, N.: Efficient Monte Carlo simulation method of GERT-type network for project management. *Comput. Ind. Eng.* **42**, 521–231 (2002)
5. Kosztyn, Z., Kiss, J.: Stochastic network planning method. In: Elleithy, K. (ed.) *Advances Techniques in Computing Sciences and Software Engineering*. Springer B.V. (2010)

6. Kutschenriter-Praszkiewicz, I.: Systemy bazujące na wiedzy w technicznym przygotowaniu produkcji części maszyn. Wydawnictwo ATH, Bielsko-Biała (2012)
7. Wang, ChN, Yang, G.K., Hung, KCh., Chang, K.H., Chu, P.: Evaluating the manufacturing capability of a lithographic area by using a novel vague GERT. *Expert Syst. Appl.* **38**, 923–932 (2011)
8. Pritsker, A.: GERT: graphical evaluation and review technique. National Aeronautics and Space Administration. Memorandum RM-4973 NASA. The RAND Corporation Santa Monica, California (1966)
9. Lu, J., Jain, L.C., Zhang, G.: *Handbook on Decision Making, Vol 2: Risk Management in Decision Making*. Springer, Berlin (2012)
10. Mathew, J., et al.: *Engineering asset management and infrastructure sustainability*. Springer, London (2011)

A Digital Pattern Approach to 3D CAD Modelling of Automotive Car Door Assembly by Using Directed Graphs

S. Patalano, F. Vitolo and A. Lanzotti

Abstract The present paper deals with methods for product development aimed to support designing activities and to re-use company know-how. The work is addressed to complex products i.e. products characterized by several components and dependencies among them. Then, the paper presents both the methodological approach and the application to the 3D CAD modelling of an automotive car door assembly. The work uses directed graphs and a series of algorithms to provide a Graphical User Interface (GUI) able to support a designer by reducing the development time of new car door assemblies and increasing the accuracy of the design activities. According to a digital pattern approach, the GUI is used to determine the set of changes to 3D CAD models that typically occur in the automotive field, during the development of new car door assemblies.

Keywords Digital pattern for product development · CAD modelling · Directed graphs · Automotive car door design and development

1 Introduction

In the automotive and railway field, advanced methods for 3D CAD modeling increasingly concern the development of complex products i.e. products characterized by several components and dependencies among them. A *top-down* approach to CAD modelling is a useful way to deal with the designing of complex products. In fact, such an approach provides criteria for breakdown structure by identifying less complex and independent sub-assemblies as to define the datum features necessary for product development [1].

S. Patalano (✉) · F. Vitolo · A. Lanzotti
Fraunhofer JL IDEAS, Department of Industrial Engineering,
University of Naples Federico II, P. le V. Tecchio 80, 80125 Naples, Italy
e-mail: stanislao.patalano@unina.it

Several researchers worked on a top-down approach to designing and modelling. Chen et al. [2] developed a multi-level assembly model that is useful to capture information and define dependencies in different levels of the decomposition. The way to integrate a *top-down* approach to designing methods was also tackled in [3], with a particular reference to subsidiary industries. A top-down approach could also be applied to easily control the degree of approximation of dimensional and geometrical modeling of turbomachinery rotor [4].

Some CAD-CAE environments provide tools to explore and manage component relations but often this task is time consuming due to the high number of components, features and related dependencies. Furthermore, some tools available in commercial CAD-CAE environments allow a simple visualization of existing dependencies providing file part names. In some cases, a graph representation is also used but no tools are available for the designer to address existing dependencies between selected components or features. In such a context, graph theory [5] and related algorithms [6] can be used to accomplish the interactive search of dependencies.

Some works in literature are aimed to integrate tools based on graph representation within CAD systems. In [7] a tool for a quick checking of tolerance specifications, directly assigned to a CAD model of an assembly, were proposed. In particular, graphs provide the right abstraction from the CAD model while preserving the accuracy of the dependencies and the interconnections between features. Lockett and Guenov [8] present an approach based on graphs for a module integrated in a CAD system and aimed to feature recognition, addressed to molded parts. In [9], a designing approach based on graph theory and a related *Graphical User Interface* (GUI), are presented. In particular, the approach allows the management of company know-how according to a *Knowledge Based Engineering* (KBE) point of view and provides an easy-to use graphical interface based on directed graphs for the designing of automotive gearboxes.

The challenge of reducing designing time for new mechanical assemblies, especially in the context of large companies, encourages the use of methods and tools aimed to support designing activities and to re-use company know-how. Furthermore, the design choices must be rapidly checked to avoid errors that could cause delay or expensive re-designing. Therefore, some large companies, operating within automotive and railway fields, have deepened the tools and methods to accomplish “digital patterns for product development” viz. geometrical data and models, to be reconfigured and re-used in different but similar design activities [10, 11]. These patterns are generally supported by a series of tools, as for example, “quality and standard checkers”, “predictive engineering wizards” and “cost advisors” that are part of a decision support system. Graph representations and a related algorithm could act as a part of such a decision support system.

The present paper tackles the methods of re-using company know-how, related to the development of automotive car-body assemblies. The paper proposes a method based on graphs to integrate company knowledge into 3D CAD models, by means of an approach to Digital Pattern (DP) for product development.

The paper is arranged as follows. Section 2 presents the DP approach to 3D CAD modelling and the role of the analysis based on graph representation; Sect. 3 provides the application to the case of an automotive car door assembly; Sect. 4 draws the Conclusions.

2 The Digital Pattern Approach to 3D CAD Modelling

The Digital Pattern (DP) approach aims to develop models, to be reconfigured and re-used in similar designing tasks. Therefore, the DP models are geometrical and numerical models, preconfigured and parametrical, able to be adapted according to the constraints of a new designing task. Then, the designer, by means of DP models, reduces the development time of new complex products and increases the accuracy of the results as the tools of a decision support system determine and suggest to the user. Specific models, international standards and company rules are used to develop such tools.

Figure 1 depicts the DP approach. In particular, the exchange of contents between *parametrical CAD models* and the *tools of the decision support system* is cyclic as it deals with the continuous development of each updated CAD model aimed to fulfill the complex product requirements.

The cyclic relationship between *parametrical CAD models* and *the tools of the decision support system* has to bring within the product/process development both the set of constraints coming from the adoption of a specific manufacturing process as well as the results of simulations performed on preliminary models i.e. design archetypes. The CAD models of a complex product, that are continuously updated, use a directed graph or *digraph* called Design Digraph (DD) as the cornerstone for

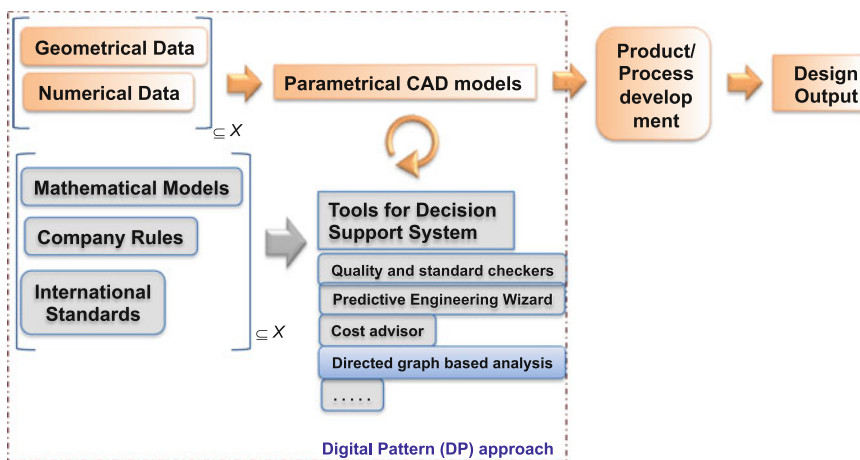


Fig. 1 Contents and operational flow of the DP approach for product/process development

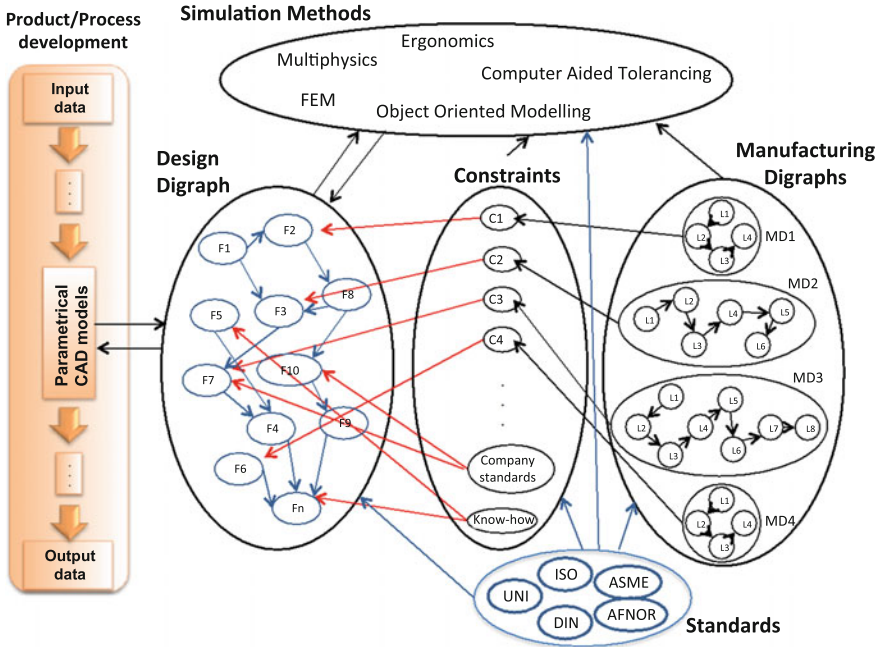


Fig. 2 Map of links existing among DD, alternative manufacturing digraphs, constraints, standards and simulation methods

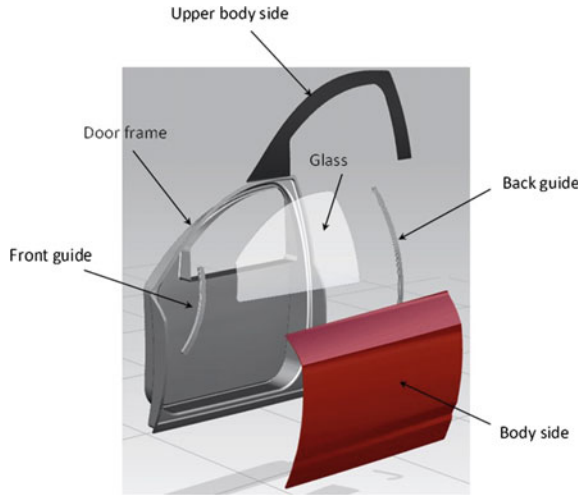
the exchange process of designing information. Figure 2 shows the links between DD, alternative Manufacturing Digraphs, Constraints, Standards and Simulation Methods.

The designing activities that have to be performed on parametrical models induce use of datum and a top-down approach as key factors for 3D CAD modelling. Designing activities, in fact, need to explore and control the consequences of a series of changes to parametrical CAD models to accomplish new and updated releases of such models, assuring the fulfillment of standards, manufacturing constraints and costs. Therefore, primary datum are usually imported as source nodes within the DD.

3 The DP of an Automotive Car Door Assembly

The DP approach was applied to the development of an automotive car door. In particular, the present Section deals with the 3D CAD modelling activities performed to accomplish the development of the automotive car door assembly. We considered a reference car door model containing the set of primary characteristics that are common to all car door assemblies. The reference car door model is

Fig. 3 Exploded model of the car door assembly used as reference



characterized by a set of parts (as depicted in Fig. 3): door frame; front and back guide; glass; body side and upper body side. Therefore, a preliminary functional decomposition could be used to address all car door parts and related assembling relationships between them.

Figure 4 shows the functional decomposition of the car door assembly by using a tree representation. In particular three levels are depicted: the first is related to the

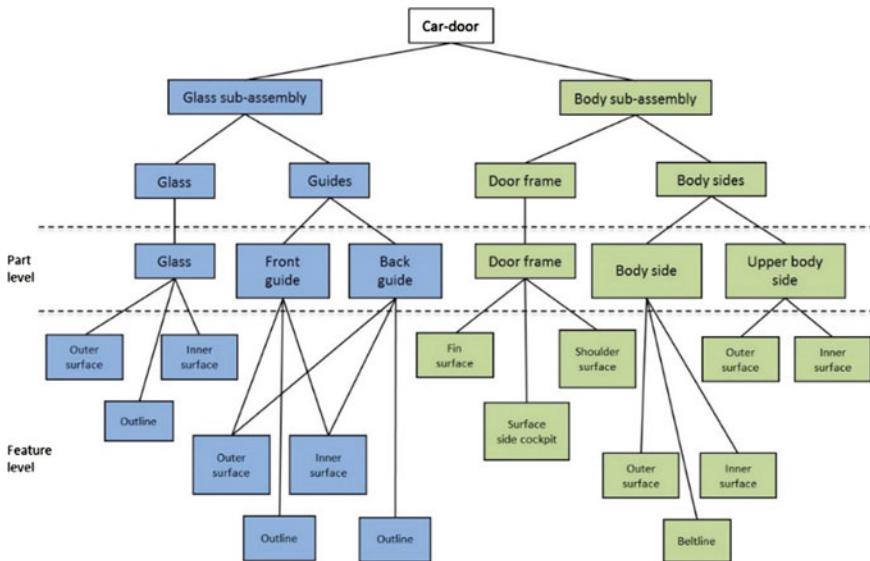


Fig. 4 Tree representation of the car door assembly

main assembly and sub-assemblies; the second is related to the part level and, finally, the third is related to the feature level.

Furthermore, the functional decomposition enables the development of graph representation related to relationships that exist among components (called *part level*) and among features (called *feature level*). Besides, also the related geometrical references i.e. datum could be represented by means of the same digraph (DD), as well as a set of significant parameters (such as glass thickness and body side thickness) addressed according to manufacturing constraints, as depicted in Fig. 5.

The nodes represent features and geometrical parameters characterizing the assembly, while directed edges represent geometrical and functional dependencies among features and parameters.

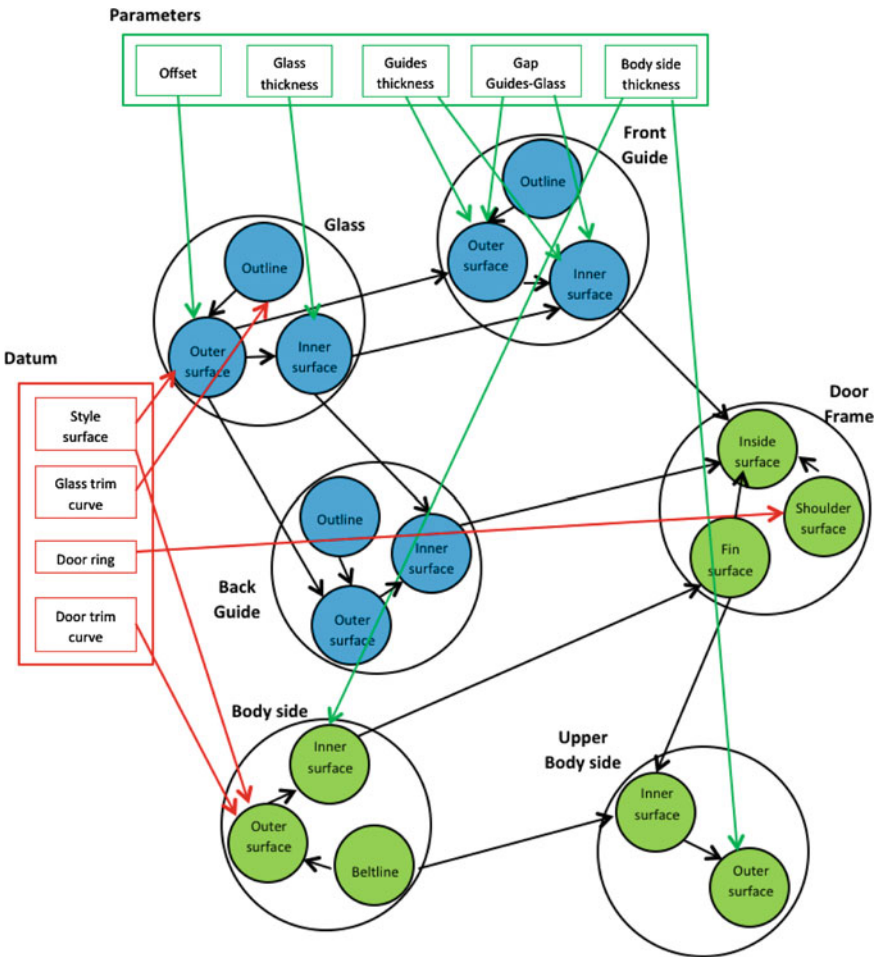


Fig. 5 Car door design digraph

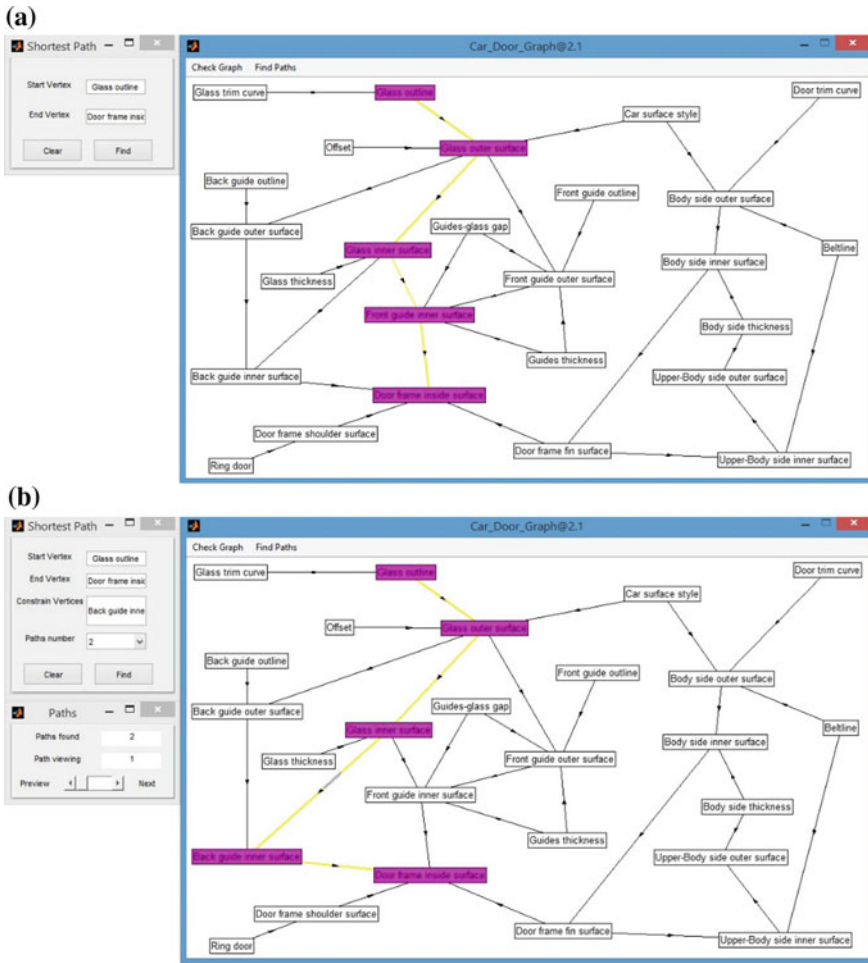


Fig. 6 GUI operating for: **a** the shortest path between two nodes; **b** the shortest path between two nodes through a constraint node

To accomplish the generation of DD and to analyze dependencies between nodes, a *graphical user interface* (GUI) was developed (Fig. 6). In particular the GUI, developed within a MATLAB environment, is able to support the designer by means of a series of algorithms. The algorithms implemented within the interface are the following:

1. The *algorithm for the shortest path* by Dijkstra [6]; the algorithm finds the shortest path by selecting two existing nodes within the DD.
2. The *algorithm for ordered paths*: the algorithm finds and sorts out in ascending order all possible paths between two selected nodes (an example of use of such algorithm is depicted in Fig. 6a).

3. The *algorithm for ordered and constrained paths*; the algorithm finds and sorts out in ascending order all possible paths, between two selected nodes, that fulfill one or more selected nodes as constraint (an example of use of such algorithm is depicted in Fig. 6b).
4. The *algorithm for the shortest path to a source node*; the algorithm finds the shortest path from a selected node to a selected source (or independent) node.
5. The *algorithm for the shortest path to source nodes*; the algorithm finds the shortest path from a selected node to every source (or independent) nodes.

The implemented algorithms, operating with matrixes associated to graphs, actually do not take into account cost functions for assembly operations as the present work tackles the impact of geometrical changes due to redesigning tasks and occurring during development processes. Typical and frequent “evolutionary” processes in automotive design, in fact, consider the same assembling operations to avoid expensive changes of equipment but needs for a long series of interventions due to changes, for example, to style surfaces, glass trim curves, door rings and door trim curves. Characteristics of assembly operations, such as times and task complexity, could be introduced and managed by algorithms in order to take into account special constraints rising within the development of innovative assembling operations.

The implemented algorithms make designers able to address the right sequence of intervention to CAD model of car door assembly. Thanks to the implemented GUI, the analysis does not depend by the complexity i.e. by the number of components of the assembly.

The GUI was used to perform a set of changes to parametric CAD model of car door assembly, according to a Digital Pattern approach, by assuring the fulfillment of all design constraints. The changes to parametric CAD model of car door assembly fit some typical needs that in the automotive field occur. In particular the changes deal with (Fig. 7): the updating of glass and slot thickness; the updating of door ring; the updating of door trim curve and glass trim curve.

3.1 Discussion

Several aspects could be pointed out taking into account the car door assembly used as reference.

Characteristics of 3D modelling: to introduce a DP approach to 3D CAD modelling the use of a top-down approach is mandatory. Only the use of a top-down approach could assure the correct propagation of changes between features.

Application of changes to parametric models: the changes imposed to parametric CAD models are automatically accomplished.

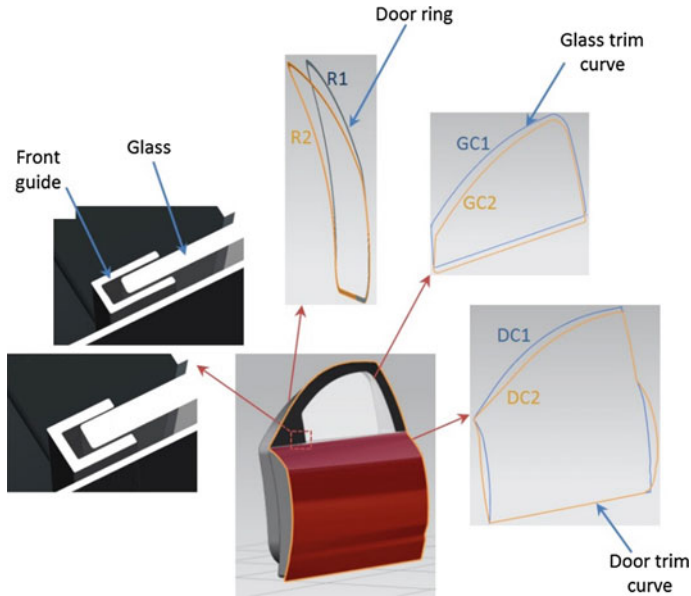


Fig. 7 a Increasing of glass and guide thickness; b updating of door ring; c updating of glass trim curve; d updating of door trim curve

GUI efficiency: GUI makes the designer able to quickly explore dependencies among different product features and, therefore, enhances the operations related to the updating of CAD models; the developed GUI demonstrates good efficiency, as the analysis of dependencies is independent by the complexity of assembly. The more complex is the assembly, the more useful is the developed GUI.

Enhancements due to graph representation: the graph representation allows pointing out the dependencies between parts or assembly features and related constraints coming from standards and manufacturing processes. Therefore, company knowledge could be continuously and easily added to graph representation and, then, to CAD models. This aspect has strategic meaning for large companies engaged in a continuous improvement of product development processes.

Increasing of assembly CAD models: the use of graphs allows the iterative increasing of CAD features to take into account further details of car door assemblies. The increasing of assembly CAD features is essential to guarantee the re-use of car door DD for the development of new releases of car doors, frequently occurring with automotive industries.

Graph generation: actually, input data of GUI are provided as arcs list, while the graph generation is automatic.

4 Conclusions

The paper presents a digital pattern approach, based on graphs, to 3D CAD modelling of automotive car door assembly. In particular, the approach proposes a directed graph based analysis as a part of a decision support system and the use of a top-down modeling to accomplish the successive steps of designing activities for product development. Therefore, the contribution of the present study is an approach to accomplish the development of CAD modelling for car door assemblies, dealing with several features and dependencies, to overcome the actual limits of tools implemented in commercial CAD software. These tools, in fact, allow only the one-by-one inspection of existing relations between features and therefore they cause time consuming tasks when features and related dependencies increase.

A GUI, implementing a set of sorting algorithms, was accomplished to support the designer during the analysis of dependencies between parts and features. The implemented GUI was tested on the case of a car door assembly, used as reference, to explore all existing dependencies among different parts and features, and to easily operate changes, in an automatic way, to CAD models. The GUI is now not embedded in any commercial CAD system to assure a wide use; on the other side the definition of arcs list needs for time and expert users. The efforts due to generate the directed graph are compensated by the possibility to use the same graph for different projects i.e. different releases of the car door assembly. Further developments deal with the GUI implementation as an add-on in a specific CAD environment. In such a way, it will be possible to directly import the dependencies coming from geometrical features and, finally, edit by means of an arcs list the dependencies due to standards and manufacturing constraints.

Acknowledgments The present work was developed with the economic support of MIUR (Italian Ministry of University and Research) performing the activities of the project PON01_01268 “Digital pattern product development: a pattern driven approach for industrial product design”.

References

1. Pimpler, T.U., Eppinger, S.D.: Integration analysis of product decompositions. In: Proceedings of ASME Design Theory and Methodology Conference, Minneapolis (1994)
2. Chen, X., Gao, S., Yang, Y., Zhang, S.: Multi-level assembly model for top-down design of mechanical products. *Comput. Aided Des.* **44**, 1033–1048 (2012)
3. Aleixos, N., Company, P., Contero, M.: Integrated modeling with top-down approach in subsidiary industries. *Comput. Ind.* **53**, 97–116 (2004)
4. Patalano S.: A systematic CAD approach for turbomachinery rotor modelling. In: CD Proceedings of the 5th International Conference on Advanced Engineering Design, Prague, Czech Republic, June. ISBN 80-86059-44-8 (2006)
5. Bondy, J.A., Murty, U.S.R.: *Graph theory*. Springer, Berlin (2008)
6. Deo, N.: *Graph theory with applications to engineering and computer science*. PHI Learning (2004)

7. Franciosa, P., Patalano, S., Riviere, A.: 3D tolerance specification: an approach for the analysis of the global consistency based on graphs. *Int. J. Interact. Des. Manuf.* **4**(1), 1–10 (2010)
8. Lockett, H.L., Guenov, M.D.: Graph-based feature recognition for injection molding based on a mid-surface approach. *Comput. Aided Des.* **37**, 251–262 (2005)
9. Patalano, S., Vitolo, F., Lanzotti, A.: A graph-based software tool for the CAD modeling of mechanical assemblies. In: *Proceedings of 8th International Conference on Computer Graphics Theory and Applications GRAPP/IVAP2013*, pp. 60–69. Barcelona (2013)
10. Lanzotti, A., Patalano, S., Vitolo, F.: A graph-based approach to CAD modeling: a digital pattern application to the sizing and modeling of manual transverse gearboxes. In: *Proceedings of International Conference on Graphic Engineering, Madrid* (2013)
11. Patalano, S., Vitolo, F., Lanzotti, A.: A digital pattern approach to design of an automotive power window by means of object-oriented modelling. In: *Submitted to International Conference on Multiphysics Modelling and Simulation for Systems Design MMSSD2014, Sousse* (2014)

Part IV
Graph-Based Aid and Modelling
of Design Tasks

Application of Game Graphs to Describe the Inverse Problem in the Designing of Mechatronic Vibrating Systems

A. Deptuła

Abstract Dependence graphs and parametric game trees can be applied to describe dynamic properties of machine systems. Discrete mechatronic systems are understood as a combination of mechanical discrete models with piezoelectric elements and external electrical circuits *LRC*. An application of game graphs to describe the inverse problem on the basis of an example of a mechatronic vibrating system has been presented in the paper, focusing on mechanic subsystems of concentrated parameters and an external *LRC* system.

Keywords Game-graph · Game-tree structures · Mechatronic systems · Cascade systems · Modelling · Synthesis · Transformation

1 Introduction

Discrete mechatronic systems are understood as a combination of mechanical discrete models with piezoelectric elements and external electrical circuits *LRC*. In order to avoid a large number of analyses, the searched sets are subject to a synthesis that is an inverse problem to an analysis [1–6]. The inverse dynamics problem of the discrete vibrating systems with the application of dimensionless transformations and retransformations has been shown in the papers [1–3]. These works are a development of research done by the scientific centre in Gliwice concerning the synthesis of mechanical systems, ways of vibration damping, drive system susceptibility and the application of piezoelectric elements in the vibration damping. It has been shown in papers [1, 2] that there is a possibility of realizing the semi-active vibration isolation by means of using the system with negative capacitance, whereas in the paper [5, 7] vibrating mechanical systems have been described by means of non-homogeneous coordinates by elaborating formalism of matrix hybrid graphs. Another approach is to analyse mechatronic structures, in

A. Deptuła (✉)
Opole University of Technology, Opole, Poland
e-mail: a.deptula@po.opole.pl

particular those composed of discrete mechanical systems by means of signal flow dependence graphs. An application of game graphs to describe the inverse problem on the basis of an example of a mechatronic vibrating system has been presented in the paper.

2 Application of Dependence Graphs and Parametric Game Trees in a Discrete Description of Mechatronic Systems

The problem of modelling mechanical systems by means of graphs of different categories is well-known [1–8]. Dependence graphs and parametric game trees can be applied to describe dynamic properties of machine systems [4, 9–12]. The mathematical model of systems is composed of a group of functions connecting different variables with each other and in this way describing connections among different variables in the system. It is possible to state mutual connections of all functions dependent on time. In the dependence graph describing the system of algebraic, differential and integral equations, functions dependent on time have been described as a vertex.

Figure 1 presents a branched system of three degrees of freedom [1].

Motion equations of the system from Fig. 3 have been presented in the following form:

$$\begin{cases} \frac{m_1}{c_1} \ddot{x}_1 = \frac{F_0}{c_1} \cos \Omega t - x_1 - \frac{c_3}{c_1} x_1 + \frac{c_3}{c_1} x_3 - \frac{c_2}{c_1} x_1 + \frac{c_2}{c_1} x_2 \\ \frac{m_2}{c_2} \ddot{x}_2 = -\frac{c_4}{c_2} x_2 - x_2 + x_1 \\ \frac{m_3}{c_3} \ddot{x}_3 = -x_3 + x_1. \end{cases} \quad (1)$$

The considered motion Eq. (1) can be written with the dependence graph for tree game structures (Fig. 2).

The graph distribution from any vertex in the first stage leads to a tree structure with cycles, and next to a general tree game structure. Each structure has a proper analytic notation G_i^{++} —where i —is a vertex, from which the graph decomposition started determining a way of transition from the dependence graph to the tree structure.

Analytical expression of game tree structures $G_{x_1}^{++}$ —from the initial vertex x_1 (2) and game tree structures $G_{x_2}^{++}$ —from the initial vertex x_2 (3):

$$G_{x_3}^{++} = \left(\ddot{x}_3 \left(\iint dx x_3 \left(\frac{c_3}{c_1} \ddot{x}_1 \left(\iint dx x_1 \left(1 \ddot{x}^2 \left(\frac{m_3}{c_3} \ddot{x}_3, \iint dx x x_3 \left(-1 \ddot{x}_3^2, \frac{c_3}{c_1} \ddot{x}_1^4 \right) \right) \right) \right) \right) \right) \right), -1 \ddot{x}_3^2, \frac{m_3}{c_3} \ddot{x}_1 \right) \quad (2)$$

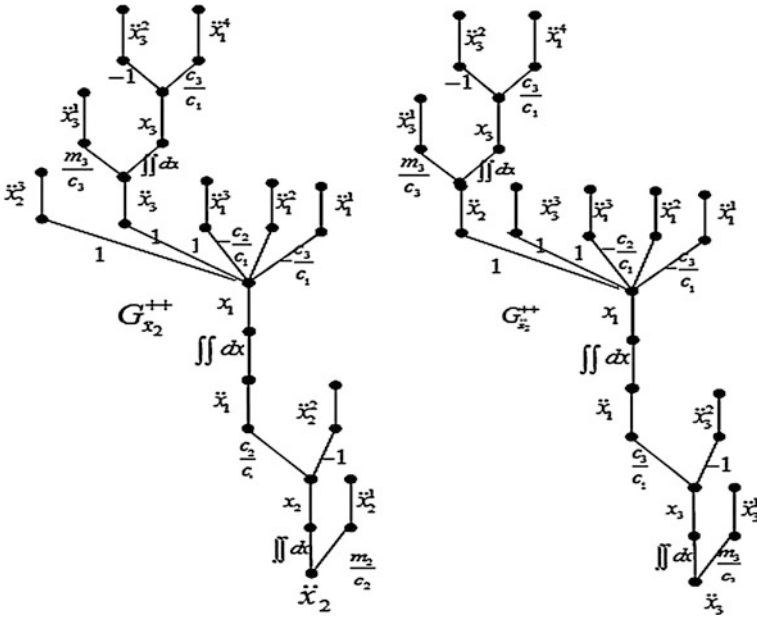


Fig. 3 Tree game structure from the initial vertex x_2, x_3

3 The Inverse Problem in the Designing of Mechatronic Vibrating Systems

Dimensional retransformations are used in order to make a required mechanical set out of the dimensionless system. In order to make a synthesis of mechatronic systems we use mainly the basic piezoelectric compounds, combining mechanical and electrical issues.

The piezoelectric element is connected to the external *LRC* system. Thanks to the *L*, *R* and *C* parameters it is possible to realise passive damping or adaptive-passive damping.

The electric charge Q_p can be expressed in the following way:

$$Q_p = e \frac{A_p}{l_p} x_i + \varepsilon_S \frac{A_p}{l_p} u_p. \tag{4}$$

The described parameters are indispensable to determine the mechatronic system, meeting the stated requirements in the form of a sequence of zeros and poles [1, 2].

In the designed systems, damping elements are used as a limited resort, in particular a piezoelectric actuator which acts as a “stack”. Piezo-electric crystal is

composed of a large number of interconnected individual piezoelectric layers where the displacement of the resultant displacement is the sum of the individual layers.

In [3] we see a synthesized vibrating mechatronic system. Characteristic functions are distributed by the extended [2], while the designed structure due to the required dynamic properties is a string of zeros and poles.

In each case designed mechatronic systems, as compared to those obtained in the first step of mechanical systems with lumped parameters, differ in the number of degrees of freedom.

Characteristic functions of stabilised systems is seen as slowness:

$$U(s) = H \frac{d_l s^l + d_{l-1} s^{l-1} + \dots + d_0}{c_k s^k + c_{k-1} s^{k-1} + \dots + c_1 s}, \tag{5}$$

where:

- l an odd degree of a numerator,
- k a degree of a denominator, $l - k = 1$,
- H any positive real number or mobility:

$$V(s) = H \frac{c_k s^k + c_{k-1} s^{k-1} + \dots + c_1 s}{d_l s^l + d_{l-1} s^{l-1} + \dots + d_0}. \tag{6}$$

When we write the λ_2 parameter in relation to the retransformed set:

$$\lambda_2 = L_x C_{ps} \omega_1^2 \text{ or } \frac{m_3}{c_3} = L_x C_{ps} \omega_1^2 \tag{7}$$

where:

- L_x inductance of the winding of the external electric circuit
- C_{ps} piezoelectric stiffness in the mechanical system replacement.

It is possible to make transformations of the matrix set of equations in the Lx configuration to [2, 3]:

$$\begin{aligned} & \begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ L_x C_{p,s} & 0 & \frac{\epsilon_S}{e} L C_{p,s} \end{bmatrix} \underbrace{\begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \ddot{u}_p \end{bmatrix}}_{\underline{W}''} + \begin{bmatrix} c_0 + c_{pm} + c_2 & -c_2 & \frac{-c_{pes}\epsilon_S}{e} \\ -c_2 & c_2 + c_4 & 0 \\ 0 & 0 & \frac{\epsilon_S}{e} \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ u_p \end{bmatrix}}_{\underline{W}} \\ & = \begin{bmatrix} F(t) \\ 0 \\ 0 \end{bmatrix} \end{aligned} \tag{8}$$

As a result of transformations, a mechatronic set composed of a discrete mechanical system combined with a piezoelectric element has been obtained out of a structure of the mechanical system obtained as a result of a synthesis (Figs. 4 and 5).

in order to obtain the required behaviour of the system (for example machine system). Unlike traditional dependence graphs and tree classifiers, the dependence graph with game tree structures includes connection of importance ranks of vertices (states) and height of the tree structure.

Application of decision decomposition for game graphs and trees does not change types and graphical shapes of such structures. They are more complex but they keep the given structural properties resulting from the initial dependence graph. Thus, a local role of decomposition can be distinguished. The introduced decision decomposition eliminates interaction of constructional and service parameters because a designer can make a decision about only single changes and observations at the successive stages.

The introduced additional time vertex resulting from the physical model well describes the feedback loop on the dependence graph. It appears from the following statements:

- an additional decision is a guideline for a designing engineer when he/she is going to change the output signal on the basis of the input signal,
- the tree structure with cycles and the tree game structure preserve identical shapes like before introduction of the additional time vertex, but they have complementary branches at suitable floors.

The introduced multiple vertex numeration allows us to consider and distinguish the same elements occurring on different floors of the tree structure. Those also allow us to preserve a general shape of the graphical structure existing even before decomposition.

It would be necessary to assign appropriate elements: $d_l s^l + d_{l-1} s^{l-1} + \dots + d_0$ and $c_k s^k + c_{k-1} s^{k-1} + \dots + c_1 s$ in the game graph to the elements $z_{iv} q_{rc}$ being in the segments $(^k \dots z_i q_r \dots)^k$ and then $\dots (^{k+1} \dots)^{k+1}$. Then, characteristic functions for stabilised systems $U(s)$ and $V(s)$ (slowness and mobility) would be analysed directly on the graph. What is more, the vertex standing for the variable dependent on time and edges should have appropriate piezoelectric compounds assigned to them.

4 Conclusions

Game structures for dimensional retransformation are more complex than tree structures as far as decision-making is concerned for structures without dimensional retransformation. However, if we applied a decision decomposition, we would obtain a more complex decision-making process on tree structures for the system of three degrees of freedom. The decision model of the graph structure for a given mechatronic system prohibits the engineer-designer of a possibility of making subjective decisions, by means of changing the value of guidelines of different construction parameters.

By means of a structure of game trees, it is possible to make structural decisions in identical sequences of construction/exploitation parameters but present on different parts of the tree, resulting from different structural, designing decisions made earlier for the dimensional retransformation.

The whole game algorithm of a parametric graph could then describe the complete transformation of the mechanical system as a method of solving an inverse problem. The game structures would then stand for the set of possible solutions (structures). The way of graph designing and forming will be the subject of forthcoming articles and publications.

References

1. Białas, K.: Passive and active elements in reduction of vibrations of torsional systems. *Mechatronic Systems and Materials, Mechatronic Systems and Robotics Book Series. Solid State Phenom.* **164**, 260–264 (2010)
2. Buchacz, A., Gałęziowski, D.: *Odwrotne zadanie dyskretnych drgających układów mechatronicznych*, Monografia, Wydawnictwo Politechniki Śląskiej. ISBN 978-83-7335-977-22012, Gliwice (2012)
3. Buchacz, A., Gałęziowski, D.: Synthesis and dimensionless transformations of mechatronic vibrating systems. In: *Dynamical Systems Analytical/Numerical Methods, Stability, Bifurcation and Chaos*, pp. 249–254 (2011) (Łódź: Wyd. Pol. Łódzkiej)
4. Buchacz, A., Wróbel, A.: *Modelowanie i badanie wpływu zjawiska piezoelektrycznego na charakterystyki układu mechatronicznego*. Wydawnictwo Politechniki Śląskiej, Gliwice (2010)
5. Zawiślak, S.: *The Graph-based Methodology as an Artificial Intelligence Aid for Mechanical Engineering Design*, Wydawnictwo Akademii Techniczno-Humanistycznej, Bielsko-Biała, ISBN 8362292962 (2010)
6. Wojnarowski, J.: *Grafy i liczby strukturalne jako modele układów mechanicznych*, Inst. Podst. Konst. Masz. Polit. Śląs., Gliwice (1986)
7. Świder, J.: *Grafy hybrydowe w modelowaniu drgających układów mechanicznych z liniowymi sprzężeniami*. Ph.D.-thesis, Politechnika Śląska, Gliwice (1981)
8. Wojnarowski, J., Buchacz, A., Nowak, A., Świder, J.: *Modelowanie drgań układów mechanicznych metodami grafów i liczb strukturalnych*, Skr. Nr 1266, Politechnika Śląska, Gliwice (1986)
9. Deptuła, A.: *Kompleksowy współczynnik złożoności dla struktur rozgrywających parametrycznie z grafu zależności przepływu sygnałów*, XLII Konf. Zast. Mat., Zakopane 2013, Inst. Mat. PAN, Warszawa (2013)
10. Deptuła, A., Łuszczyna, R., Partyka, M.A.: *Zastosowanie graficznych struktur decyzyjnych w metodologii projektowania i zarządzania na przykładzie CAD układów maszynowych*, Studia i Monografie, Zeszyt nr 315, Oficyna Wydawnicza Politechniki Opolskiej, Opole. ISBN 978-83-62736-68-3 (2012)
11. Deptuła, A., Partyka, M.A.: *Application of game graphs in optimization of dynamic system structures*. *Int. J. Appl. Mech. Eng.* **15**(3), 647–656 (2010)
12. Deptuła, A.: *Synteza grafów zależności dla struktur drzewiastych rozgrywających parametrycznie*, XLIII Konf. Zast. Mat., Zakopane 2014, Inst. Mat. PAN, Warszawa (2014)
13. Deptuła, A., Partyka, M.A.: *Badanie własności dynamicznych układów maszynowych z uwzględnieniem wielokrotnej numeracji wierzchołkowej dla drzew rozgrywających parametrycznie; Napędy i Sterowanie*, 3 (2010)

14. Deptuła, A.: Determination of game-tree structures complexity level in discrete optimization of machine systems. In: International Masaryk Conference for Ph.D. students and young researches, December 12–16, Hradec Kralove, Czech Republic (2011)
15. Deptuła, A.: Application of multi-valued weighting logical functions in the analysis of a degree of importance of construction parameters on the example of hydraulic valves. *Int. J. Appl. Mech. Eng.* **19**(3), 539–548 (2014). ISSN 1425-1654 (University Press Zielona Góra)

Graphic Matrix Formalization of Logical Decision Trees in the Optimization of Machine Systems

A. Deptuła and M.A. Partyka

Abstract Multi-valued logical decision trees indicate the importance rank of construction and/or exploitation parameters. There is an isomorphic interpretation of logical transformations, thus the Quine–McCluskey algorithm of the minimization of individual multi-valued logical functions can be considered by taking into account the graphic matrix formalization in the optimization of machine systems.

Keywords Matrix formalization · Multi-valued decision trees · Quine–McCluskey algorithm · Importance rank of decision parameters · Optimization

1 Introduction

The method of minimization of complex, partial, multi-valued logical functions indicates the importance rank of construction and/or exploitation parameters playing the role of logical decision variables [1–3].

Fluid-flow machines form a vast group of sets used in industry [3, 4]. Decision tables and logical functions [1, 4, 5] can be applied in the issues of modelling machine systems with differential equations (ordinary and partial ones). It results from the fact that non-linear elements can be divided into a finite number of linear elements (parts) which leads to getting several linear systems. Discrete optimization of fluid-flow machines is based on indicating the degree of importance of construction and exploitation parameters. Guidelines concerning the sequence of making decisions result from multi-valued decision trees by taking into consideration the realisation of the assumed purpose function (e.g. the system stabilisation).

All transformations concern the so-called Quine–McCluskey algorithm of the minimization of individual partial multi-valued logical functions. There is an iso-

A. Deptuła (✉) · M.A. Partyka
Opole University of Technology, Opole, Poland
e-mail: a.deptula@po.opole.pl

M.A. Partyka
e-mail: m.partyka@po.opole.pl

morphic interpretation of logical transformations, thus the Quine–McCluskey algorithm of the minimization of individual partial multi-valued logical functions can be considered by taking into account the graphic matrix formalization. An application of the matrix formalization in case of a bigger number of decision variables (as construction and/or exploitation parameters) makes it possible to solve practical geometric problems in order to extract the most and the least important data [6].

2 Graphic Matrix Formalization of Logical Decision Trees

A logical tree is a structural presentation of a logical function, written in the form of a sum of products, where every element is the realisation of one solution and each component in the product is a logic variable [2]. Complex partial multi-valued logical functions state the degree of importance of logic variables, by means of changing the logical tree levels, from the most important ones (near the root) to the least important (in the upper part) because there is a generalisation of a bivalent indicator of quality into a multi-valued one; $(C_k - k_i m_i) + (k_i + K_i)$, where C_k —the number of branches k -th level, k_i —times simplification on the k -th level m_i —valued variable, K_i —number of branches of $(k - 1)$ -th level, from which branches of k -th level which cannot be simplified were created. All transformations are described by the so-called Quine–McCluskey algorithm of minimization of individual partial multi-valued logical functions.

2.1 Example

For a multiple-valued logical function $f(x_1, x_2, x_3)$, where $x_1, x_2, x_3 = 0, 1, 2$, written by means of numbers KAPN (Canonical Alternative Normal Form): 100, 010, 002, 020, 101, 110, 021, 102, 210, 111, 201, 120, 022, 112, 211, 121, 212, 221, 122, there is one MZAPN (Minimal Complex Alternative Normal Form) after the application of the Quine–McCluskey algorithm based on the minimization of individual partial multi-valued logical functions having 13 literals [2]:

$$f(x_1, x_2, x_3) = j_0(x_1)(j_0(x_2)j_2(x_3) + j_1(x_2)j_0(x_3) + j_2(x_2) + j_1(x_1) + j_2(x_1))(j_0(x_2)j_1(x_3) + j_1(x_2) + j_2(x_2)j_1(x_3))). \quad (1)$$

Figure 1 shows MAPN of a given multi-valued logical function.

In the isomorphic interpretation of the Quine–McCluskey algorithm three steps are taken for the graphic matrix formalization:

- (a) putting decision (m_1, \dots, m_n) -valued variables x_1, \dots, x_n in a certain order; creating the $n!$ primary matrices, relative to all combinations of variables,

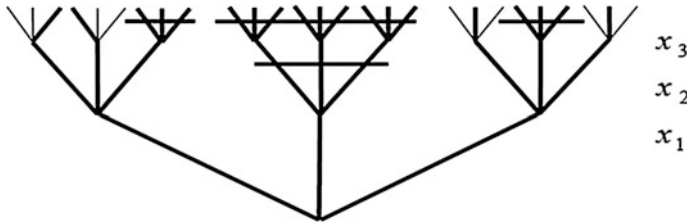


Fig. 1 A multi-valued decision tree for the parameters x_1, x_2, x_3 with an appropriate layout of levels

- (b) prioritising the numbers relative to (m_1, \dots, m_n) the valence in the increasing order from the left side of the matrix,
- (c) combining numbers and removing them (minimization).

Figure 2 shows the matrix formalization for the tree from Fig. 1.

3 Application of the Graphic Matrix Formalization in the Optimization of Machine Systems

Logical functions are taken into consideration in the tasks concerning modelling of machine sets described among others by means of differential equations (ordinary and partial ones). If a machine system is described by means of a set of functions $f_1, f_2, f_3, \dots, f_n$, dependent on time t , the change of construction parameters x_1, \dots, x_n implies a (good or bad) change in the plot of such functions. It is admissible to decrease, increase or leave the figures of construction parameters unchanged in the optimization process.

3.1 Analysis of a Degree of Importance of Construction Parameters of the Overflow Valve

The overflow valve is applied in systems in order to let excess fluid flow to the container where the pump efficacy is bigger than the need. An example of a drive system of an actuator with an overflow valve is presented in Fig. 3.

The equation of forces acting on the closing component of a valve is presented in the following way [6]:

$$\frac{Q_p^2}{A_1} \rho + \rho \cdot A_2 + \rho \cdot l \frac{dQ_p}{dt} = G_{ap} + S + k \cdot x + f \frac{dx}{dt} + m \frac{d^2x}{dt^2} + \Phi \sqrt{2 \cdot \rho} \cdot \cos(\nu) \cdot Q_p \sqrt{p} \tag{2}$$

(a)

(x_3)	j_2	j_0	j_{θ}	j_{+}	j_{-}	j_{θ}	j_{+}	j_{-}	j_{θ}	j_{+}	j_{-}	j_{θ}	j_{+}	j_{-}	j_1	j_{θ}	j_{+}	j_{-}	j_1
(x_2)	j_0	j_1		j_2			j_{θ}		j_{+}		j_{-}			j_0		j_1		j_2	
(x_1)	j_0					j_1					j_2								

(b)

(x_3)	j_2	j_0													j_1				j_1
(x_2)	j_0	j_1		j_2										j_0		j_1		j_2	
(x_1)	j_0					j_1					j_2								

Fig. 2 Matrix formalization for the decision tree from Fig. 1: **a** prioritising and combining numbers, **b** removing numbers from the table

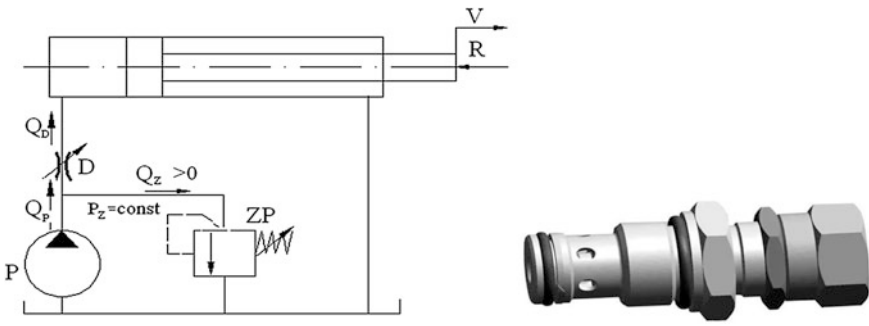


Fig. 3 A scheme of an actuator and a view of the valve

while equations of flows have the following form:

$$Q = \mu \cdot K \cdot x\sqrt{p} + A_1 \frac{dx}{dt} + \frac{V}{B} \frac{dp}{dt} \tag{3}$$

$$Q_p = \mu \cdot K \cdot x\sqrt{p} + A_1 \frac{dx}{dt} \tag{4}$$

where:

$$K = \pi \cdot d_m \sqrt{\frac{2}{\rho}} \tag{5}$$

Equations of the valve work in a dimensionless form used to make simulation are presented in the following way:

$$\rho \frac{Q_o^2}{A_1 S_o} Q_{pw}^2 + \frac{A_2 p_o}{S_o} p_w + \frac{T_{Qp}}{T_o} \frac{dQ_{pw}}{dt_w} = 1 + \frac{kx_o}{S_o} x_w + \frac{T_f}{T_o} \frac{dx}{dt_w} + \left(\frac{T_{ms}}{T_o}\right)^2 \frac{d^2x}{dt_w^2} + \Phi \frac{\sqrt{2\rho}}{S_o} \cos(\nu) Q_o Q_{pw} \sqrt{p_o} \sqrt{p_w}$$
(6)

$$Q_w = \mu x \sqrt{p_w} + \frac{T_A}{T_o} \frac{dx}{dt_w} + \frac{dp_w}{dt_w}$$
(7)

$$Q_{pw} = \mu x \sqrt{p_w} + \frac{T_A}{T_o} \frac{dx}{dt_w}$$
(8)

In order to make a discrete optimization, changes in parameters have been coded in the following way: 0- large decrease, 1- small decrease, 2- without changes, 3- increase, 4- large increase (for *m* and *k*) and : 0- small decrease, 1- without changes, 2- increase (for *d*). Depending on the adopted combinations of code changes in parameters *m*, *k* and *d* in canonical products, the behaviour of functions that depend on time is different.

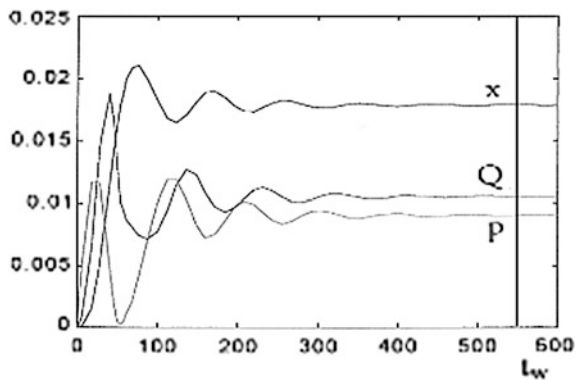
When we look at the behaviour of the functions *x*, *Q* and *p* at the time of stabilisation $t_w < 300 t_0$, we see 20 charts that have been chosen for which code changes of construction parameters *m*, *k* and *d* are presented (Table 1).

Figure 4 shows exemplary charts of the functions *x*, *Q* and *p* for the code changes of parameters (*m*, *k* and *d*) 332.

Table 1 KAPN (Canonical Alternative Normal Form) for given changes of parameters *m*, *k* and *d* ($t_w < 300 t_0$)

<i>m k d</i>	<i>m k d</i>	<i>m k d</i>	<i>m k d</i>
2 2 2	1 1 2	0 2 2	0 0 2
2 1 2	1 3 2	0 3 2	3 2 2
1 2 1	2 0 2	0 1 1	3 1 2
1 2 2	1 0 2	0 1 2	0 3 1
1 1 1	0 2 1	0 0 1	0 1 0

Fig. 4 Characteristics of the valve work for changes in the code of parameters *m*, *k* and *d*: 332



<i>k</i>	1	0	1	2	3	1	2	0	1	2	3	0	1	2	3	0	1	2	1	2
<i>m</i>	0	0			1	0			1			2								
<i>d</i>	0	1				2														
<i>m</i>	0	0	0	1	0	1	0	0	1	2	0	1	2	3	0	1	2	3	0	1
<i>k</i>	1	0	1	2	3	0			1			2			3					
<i>d</i>	0	1				2														
<i>k</i>	1	0	1	2	3	0	1	2	3	1	2	0	1	2	3	1	2	3	1	2
<i>d</i>	0	1			2			1	2			2			2					
<i>m</i>	0				1			2			3									
<i>m</i>	0	0	1	2	0	0	1	0	1	2	3	0	1	0	1	2	3	0	0	1
<i>d</i>	1	2		0	1	2			1	2			1	2			1	2		
<i>k</i>	0		1			2			3											
<i>d</i>	1	2	0	1	2	1	2	2	1	2	1	2	2	2	2	2	2	2	2	2
<i>k</i>	0	1		2	3	0	1	2	3	0	1	2	3	0	1	2	1	2		
<i>m</i>	0				1			2			2									
<i>d</i>	1	2	2	2	0	1	2	2	2	1	2	1	2	2	2	1	2	2		
<i>m</i>	0	1	2	0			1	2	3	0	1	2	3	0	1					
<i>k</i>	0		1			2			3											

Fig. 5 Matrix formalization with prioritising and combining numbers from Table 1

Each of the *canonical alternative normal form* products for appropriate changes in the value of parameters *m*, *k*, *d* (from Table 1), has 6 combinations of the matrix formalization assigned [7, 8]. Having prioritised and combined numbers, we obtain a real importance rank of parameters *m*, *k* and *d*.

Figure 5 presents a matrix formalization with prioritising and combining numbers from Table 1.

Having considered combining numbers from Table 1, we obtain the minimum form of the graphic matrix formalization presented in Fig. 6.

The minimum graphic form of the matrix formalization from Fig. 6 indicates the importance rank of construction and/or exploitation parameters from the most important ones at the bottom to the least important ones at the top.

<i>k</i>	1					1	2								0	1	2	1	2
<i>m</i>	0	0			1	0			1			2		3					
<i>d</i>	0	1				2													

Fig. 6 The minimum form of the graphic matrix formalization from Fig. 5

Table 2 KAPN (Canonical Alternative Normal Form) for given changes of parameters *m*, *k* and *d* ($t_w < 1000 t_0$)

<i>m k d</i>	<i>m k d</i>	<i>m k d</i>	<i>m k d</i>	<i>m k d</i>
2 2 2	3 3 2	0 1 1	0 3 1	2 1 1
2 1 2	1 2 1	0 1 2	0 1 0	2 0 1
1 2 2	2 0 2	0 0 1	2 3 2	1 0 1
1 1 1	1 0 2	0 0 2	3 0 2	4 2 2
1 1 2	0 2 1	3 2 2	0 2 0	4 1 2
1 3 2	0 2 2	3 1 2	2 2 1	0 0 0
1 0 0	0 3 2	1 1 0	4 0 2	0 3 0
2 4 2	4 3 2			

<i>m</i>	01	01	0	0	012	012	012	0	01234	01234	01234	01234	2
<i>k</i>	0	1	2	3	0	1	2	3	0	1	2	3	4
<i>d</i>	0				1				2				

Fig. 7 The minimum graphic form of the matrix formalization from Table 2

When we look at the behaviour of the functions *x*, *Q* and *p* at the time of stabilisation $t_w < 100 t_0$, 20 charts have been chosen for which code changes of construction parameters *m*, *k* and *d* are presented (Table 2).

Figure 7 presents the minimum form of the matrix formalization for Table 2 indicating the rank of importance of parameters *m*, *k* and *d*.

Figure 8 presents an optimal logical tree equivalent to the minimum form of the matrix formalization from Fig. 7.

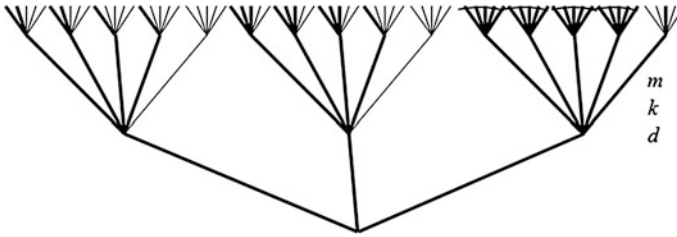


Fig. 8 An optimal multi-valued logical tree of parameters d , k , m as an equivalent of the minimum matrix formalization

4 Conclusions

Modelling examinations aim at rotating important parameters in order to ensure the stability of a real set. In the model verification it is important to state the rank of importance of construction and/or exploitation parameters [9, 10]. The paper presents an isomorphic interpretation of logical transformations of a Quine–McCluskey algorithm of minimizing individual partial multi-valued logical functions. Using graphic matrix formalization determined a degree of importance of construction parameters of the overflow valve. The complexity of logic tables or truth tables *grows* exponentially in relation to the number of variables. In case of a bigger number of decision variables, there are practical geometric problems in order to extract the least and the most important data. What is more, the graphic matrix formalization can be a computer record of parametric game trees as an adjacency matrix [5].

References

1. Deptuła, A.: Application of multi-valued weighting logical functions in the analysis of a degree of importance of construction parameters on the example of hydraulic valves. *Int. J. Appl. Mech. Eng.* **19**(3), 539–548 (2014)
2. Partyka, M.A.: Algorytm Quine’a- McCluskeya minimalizacji indywidualnych cząstkowych wielowartościowych funkcji logicznych, *Studia i Monografie Nr 109*, Politechnika Opolska – Oficyna Wydawnicza, Opole (1999)
3. Berghofer, S., Reiter, M.: Formalizing the logic-automaton connection. In: *Proceedings of the 22nd International Conference on Theorem Proving in Higher Order Logics*, LNCS No 5674, 147–163 (2009)
4. Deptuła, A., Osiński, P., Partyka, M.A.: Discrete optimization of a gear pump after tooth root undercutting by means of multi-valued logic trees. *Arch. Civil Mech. Eng.* **13**(4), 422–431 (2013)
5. Deptuła, A., Partyka, M.A.: Application of game graphs in optimization of dynamic system structures. *Int. J. Appl. Mech. Eng.* **15**(3), 647–656 (2010)

6. Deptuła, A., Partyka, M.A.: Rozłączna analiza logiczna wytycznych projektowania z uwzględnieniem niepewności w modelowaniu układów maszynowych, *J Transdisciplinary Syst. Sci.* **16**(1), 161–173. ISSN 1427-275X (2012)
7. Partyka, M.A.: Optymalizacja dyskretna pompy wirowej śmigłowej w ruchu turbinowym – zastosowanie wielowartościowych drzew logicznych. *Napędy i Sterowanie* **1** (2004)
8. Pijls, W., de Bruin, A.: Game tree algorithms and solution trees. *Theor. Comput. Sci.* 252(1-2), February, 197-215 (2001)
9. Shwnoy, P.P.: Game trees for decision analysis *Theory Decis* **44**(2), 149–171 (1998)
10. Shiue, Y.R., Guh, R.S.: The optimization of attribute selection in decision tree-based production control systems. *Int. J. Adv. Manuf. Technol.* **28**, 737–746 (2006)

The Class of Objects Graph Model as Dataware of Structural Synthesis System

O.V. Malina and E.G. Zarifullina

Abstract The article is devoted to a graph method of dataware that will form a system of construction spiroid gearboxes, based on search algorithms. Stages of the data are considered that form, from existing structures, an analysis to generalize graph structure synthesis. It is an informational base for further synthesis of new structures.

Keywords Tree graph · Structure forming modules · Structure completeness modules · Generalized model · Structural synthesis · Forbidden variant · Forbidden figure

1 Introduction

One of the most weakly formalized problem in the process of creating an engineering product is the task of structural synthesis, for example, creation of structure of a product or the technology of its manufacture.

Most universal automation engineering systems allow checking a design that an engineer selected, making calculations, or creating technical documentations. However, existing systems cannot propose a structure of a product according to the specified requirements specification.

This situation is due to a high level of creative labor that used for creation of a future product structure that is not subject to total formalization.

One way to overcome this difficulty and to formalize the process of structural synthesis is to use the unit of synthesis of combinatorial enumeration on a set of parts, components of parts and their characteristics, that called “modules”:

O.V. Malina (✉) · E.G. Zarifullina
M.T. Kalashnikov Izhevsk State Technical University, Izhevsk, Russia
e-mail: malina_0705@mail.ru

E.G. Zarifullina
e-mail: zarifullina@gmail.com

$$F_{ps} = \prod_{i=1}^{K_1} (M) \setminus (F_z \cup F_{ms}), \tag{1}$$

where: M —set of modules of objects class, K_1 —power of M set, F_z —the set of unrealizable (forbidden) variants of the object, F_{ms} —the set of identical variants of the object.

Thus, the first step in solving the problem of structural process synthesis is forming the set of modules where the enumeration will be done.

2 Preparation of the Initial Set of Modules

An analysis of existing structures of a class of objects is performed for obtainment of the set. For that we create a graph for each structure of the objects by the usual separation into “small” units (as it does constructors): subassemblies, assemblies, subassemblies, parts, etc. Graph models are widely used in research and design of mechanical engineering, in particular gearboxes [1, 2, 5, 12, 13], using tree graphs to describe the structure of objects as had been proposed by Polovinkin [11].

Graphs, which obtained decomposition as a result of various structures, have the following common features:

1. We use tree graph, whose root is the product in whole.
2. The set of the graph nodes is divided into two subsets: structure building nodes (units, details, elements of the details) (Fig. 1) and the nodes that ensure

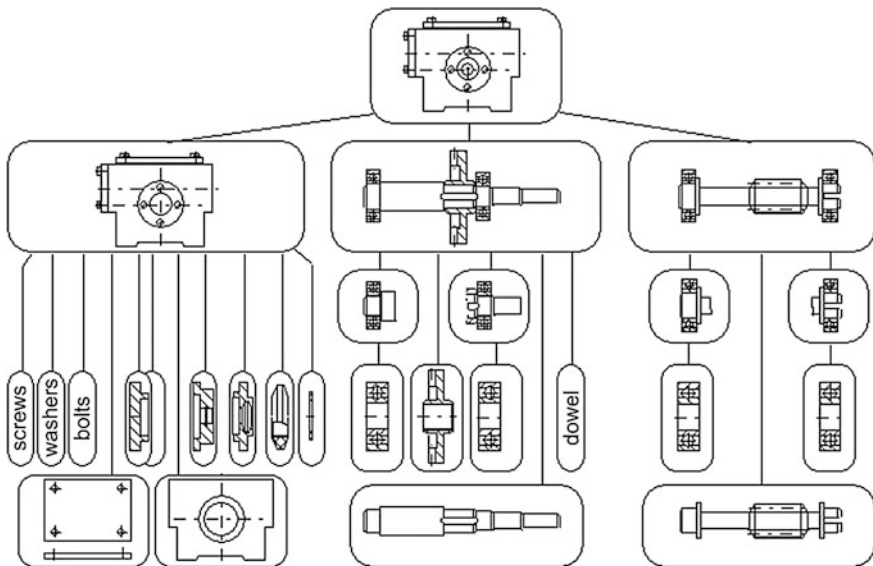


Fig. 1 The fragment of the tree graph of construction of a gear with the structure building nodes

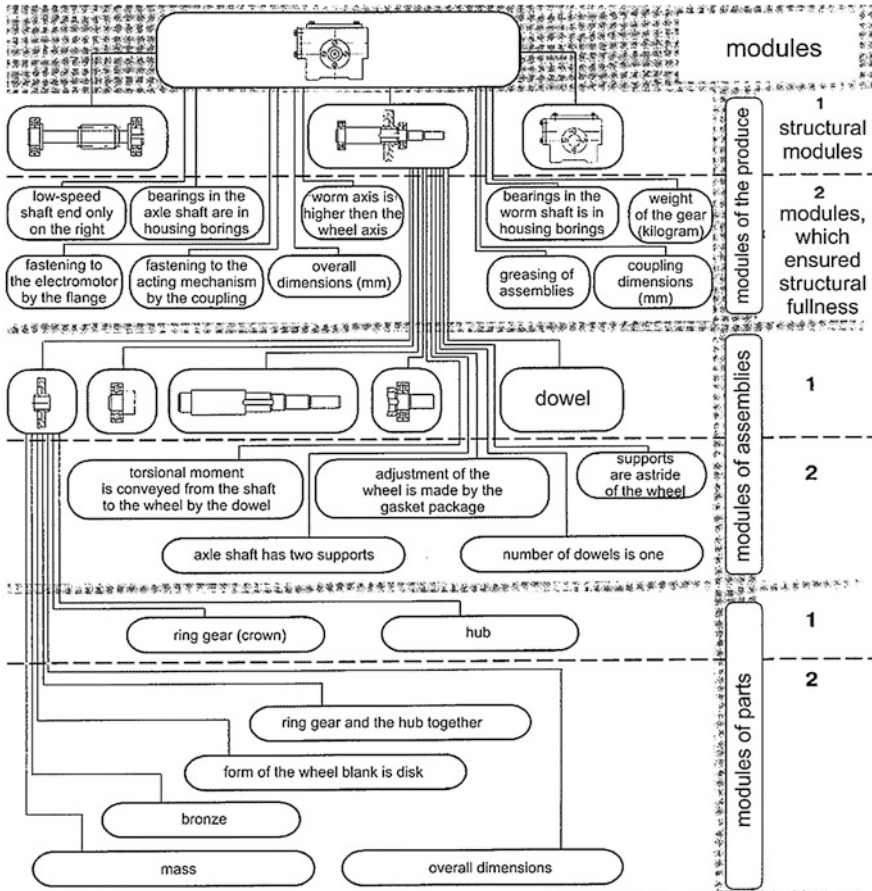


Fig. 2 The fragment of the tree graph of construction of a gear, which is complemented by structural nodes

structural completeness (size, material, consumer parameters, technical specifications) (Fig. 2).

3. Ribs of the graph point to the process of decomposition (description) of the original structure.
4. Phantom-nodes can be found in the set of structure-making nodes of the graph. Phantoms are two or more nodes that showing the same structural element. For example, the output shaft of the first stage of a two-stage gearbox and input shaft of the second stage will be displayed in the graph model as various nodes, but in reality is one functional element (Fig. 3).

Joint consideration of individual graph models, which are results of objects decomposition, allow us to draw the following conclusions:

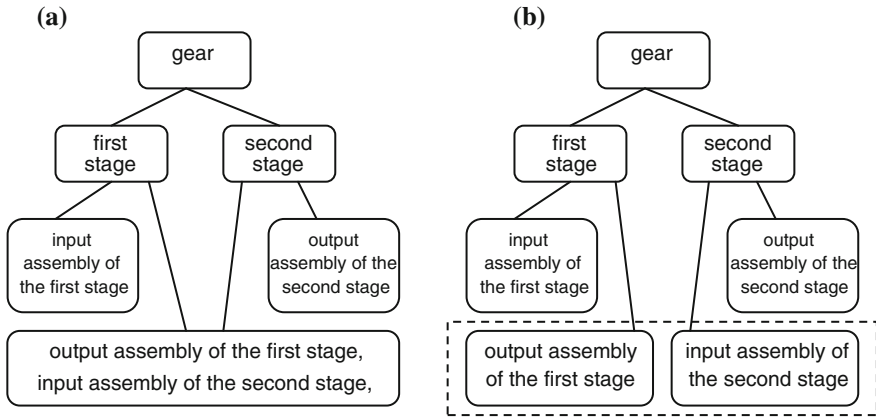


Fig. 3 **a** Real view of the gear structure; **b** hierarchical view of the gear structure, where the node “input node of the second stage” is the phantom

1. The resultant after decomposition the graphs are convenient and provide a clear view of the structures of existing designs of the product, but no one can claim to use it as a model for the entire class of objects.
2. Analysis of the set of structure building nodes allows us to draw a conclusion that some of them are obligatory (i.e. all of them exist in all graphs) and some—optional.
3. Analysis of the set of nodes that ensured structural completeness allows us to conclude that some of the nodes describing some feature of the structure building node in different graphs can have different values, so they are alternative.
4. In addition, the nodes of structural completeness may be subjected to description (characterization).
5. None of the resulting graphs of individual structures can be used as a model for the entire class of objects.

To get an object class model, generalization of the data must be produced. Then we supplement them with the extended the set of nodes.

3 Synthesis Algorithm of Generalized Model

The model with the highest plentitude of data about an objects class was suggested by Polovinkin [11]. The model is based on a comparison of various structures of models of the object, and presented by the AND-OR tree. In the tree, structural elements and features that build the structure of the construct are united by “AND” nodes. Alternative structural elements or features are united by “OR” nodes.

The model of the object class, described above, has a disadvantage. Functional elements united by the “OR” nodes supposes designing an object by bypass of a tree graph structure. Taking into account the multiple branching structures of complex objects, such an approach means high labor intensity of the process of synthesis. For solving this disadvantage a new model of object class has been suggested. According to the new model only feature-nodes can be united by the “OR” nodes. Let’s consider the graph method of constructing such a generalized model.

The generalized model construction method is based on uniting of graphs of individual variants of object. The generalized model can be obtained in the “AND-OR” tree form by semantic superposition of graphs of individual variants. It takes the following steps [4, 6]:

1. The tree-graphs intersection of individual variants

$$G_0(V_0, E_0) = \bigcap_{i=1}^N G_i(V_i, E_i), \tag{2}$$

where: N is number of initially analyzed variants of structures.

2. Analysis of the set V_0 for localization the obligatory part V'_0 :
 - if from the root node of the resulting graph v_0 , meaning the object on the whole, to the node v_i , where $i = \overline{1..K}$ ($K = |V_0 - 1|$) exist the route, then $V'_{0j+1} = V'_{0j} \cup \{v_i\}$, where V'_{0j} and V'_{0j+1} —two successive states of the pseudo-obligatory nodes, and $V'_{01} = \{v_0\}$.

Thus, the subgraph is the pseudo-obligatory part of a structure of any variant of the class of objects (skeleton of the generalized tree graph). The set of nodes of the subgraph is a connectedness component, containing the root node.

3. The set of derived nodes is given to an expert to make an analysis. As a result we get obligatory components. During the analysis an expert must answer the question “Does all known structure of the object always contain the analyzed structural unit?” A positive answer allows us to refer the node to the obligatory nodes set. Finally, it allows us to get $G'_0(V'_0, E'_0) \subseteq G_0(V_0, E_0)$. A negative answer indicates that the node is supposed to be an alternative of the first kind. Appearance of first kind alternative nodes is the result of the base sample of objects. The first kind of alternatives are excluded from the connectedness component containing the root node, and marked as “NULL”-alternative.
4. Addition of a generalized tree graph by alternative component means forming of the other (remainder) alternative nodes. It begins with the consideration of the root node and is an iterative process. Each iteration means executing the following procedures by a generalized graph from level to level:

- for the initial graphs, those numbers are fixed in the concerned node of the generalized tree graph, (if the node is obligatory or the first kind alternative, so for all of the initial graphs) are constructed subgraph $G_{lik}(V_{lik}, E_{lik})$. Set of nodes of this subgraph include the considered l -th node of the i -th level of v_{li} generalized tree graph and all adjacent with it nodes level of $i + 1$ of the initial graph;
- it is formed sets $V_{lik}^s = V_{lik} \setminus V_0' \setminus V_s$, where $V_s = \{v_{li}\}$, composed of not obligatory nodes;
- as a result of comparative analysis of the sets V_{lik}^s and its consistent viewing opposite (relative to their content) nodes have being found. And if we find at least one pair of nodes, we unite them through the node “OR” and transfer to the generalized tree graph to the $i + 1$ level. At the same time they are bounded with v_{li} node and joined into the set $V_{i0}'' \subset V_0''$, where V_0'' - the set of all nodes, united by the node “OR”. Each node is marked numbers of graphs to the set of which it belongs. If in the set of alternative nodes weren't fixed all numbers of all analyzed graphs or the parent node is marked by “NULL”, then the set of such alternatives is called incomplete and is supplemented by alternative “NULL” (it is $\overline{\bigvee_i (v_{i+1})}$, where v_{i+1} —the values of alternatives of the incomplete set). It is done for correct representation of the model;
- the other nodes ($\bigcup_{k=1}^N V_{lik} \setminus V_0' \setminus V_{i0}''$) have being included in a set V_0''' , which is called the set of supposed alternatives. At the same time the first kind of supposed alternatives together with the numbers of all graphs taken part in the scrutiny marked as “NULL”. Second kind supposed alternatives are marked only by the numbers of the applicable graphs. A generalized tree graph is admitted as ready-built (finished) when the analysis is made for all nodes (Figs. 1a–c and 4).

The finished generalized model is characterized by compactness of information storage about a large quantity of structures of the object variants and can be used to produce new variants. However, such a model has a disadvantage. It is redundancy of stored information, because during the creation of the product only unique units of object (that distinguishes one object from another) are significant. For overcoming of redundancy Eq. (1) should be converted to the following form [4]:

$$F_{pg} = M_{om} \times \prod_{i=1}^{K_2} (M_{im}) \setminus (F_z \cup F_{m6}), \quad (3)$$

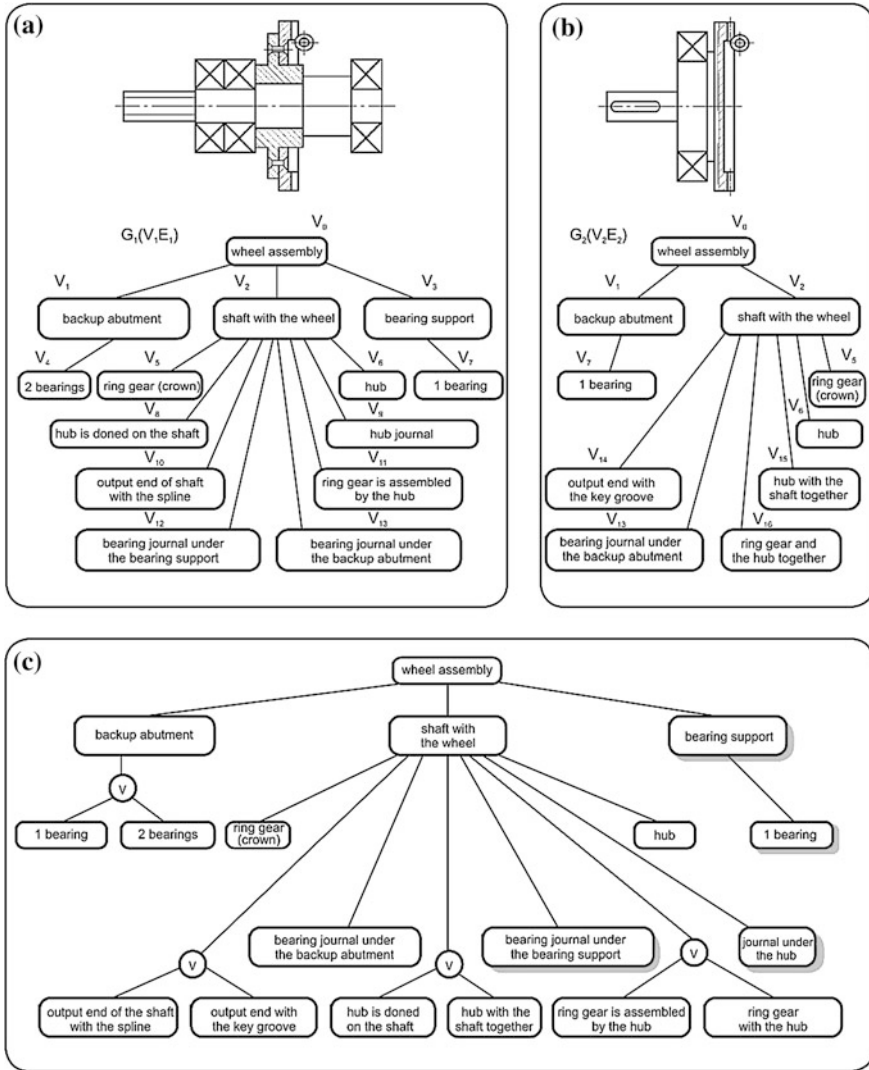


Fig. 4 Example of forming the generalized tree graph

where M_{OM} —the set of obligatory modules of the object, M_{HM} —the set of not obligatory modules, K_2 —power of the M_{HM} set.

Equation (3) is reformed into the form 3 by partitioning the set of non-obligatory modules into subsets p_i , where p_i —is the sign, which representing a set of values-modules, united by one node “OR”:

$$F_{p\beta} = M_{om} \times \prod_{i=1}^N (p_i) \setminus F_z, \quad (4)$$

where N —power of the signs set, p_i —is the i -th sign, F_z —the set of unrealizable (forbidden) variants of the object.

4 Further Steps

Further work with the resulting graph model involves building a classifier, its addition and use as an information model for the process of structural synthesis of engineering products by optimized exceeding algorithms, as it is presented in [3, 7–10].

The classifier, built on the basis of a generalized graph model, is a set of classification criteria. The criteria are formulated as questions, the answers to which actually constitute the description of a specific variant of spiroid gearbox. In other words, the classifier is the set of features describing each structure-forming module and set of their values. The result of all features, describing the class of objects, multiplication is the set of all sets that describe the structures. After exclusion of impossible (unrealizable) sets we have possible sets, including new technical solutions.

This set can be expressed by transforming the formula 1 as follows:

$$F_{p\beta} = M_{om} \times \prod_{i=1}^N (p_i) \setminus F_z = M_{om} \times \prod_{i=1, j=1}^{N, K_i} (a_j^i) \setminus F_z, \quad (5)$$

where K_i —power of set of the i -th feature values.

Despite the fact that formula (2) describes the set of all possible structures and demonstrates basic steps of its getting (Cartesian multiplication of sets of values of features and after removing forbidden variants) we cannot use it as model for the synthesis, because of the curse of dimensionality problem (information explosion appearing within the process of multiplying).

Optimization is a necessary condition for implementation of the synthesis process. In our research it is alternation of Cartesian multiplication process and exceptions structures that later would give rise to forbidden variants.

The goal of optimization is minimization of synthesis time and storage space for temporary variants.

Synthesis algorithms help to optimize. They allow identifying the causes (forbidden figures) of forbidden variants appearance. Then we use forbidden figures for removing potentially forbidden variants before their appearance. Thereby, on the one hand, we reduce the number of intermediate variants, on the other hand, saving

time spent for the process of following Cartesian multiplication and checking of presence of forbidden figures.

References

1. Drewniak, J., Kopeć, J., Zawiślak, S.: Graph models of automobile gears—kinematics. *Int. J. Appl. Mech. Eng.* **19**(3), 563–573 (2014)
2. Drewniak, J., Zawiślak, S.: Linear-graph and contour-graph-based models of planetary gears. *J. Theor. Appl. Mech.* **48**(2), 415–433 (2010)
3. Goldfarb, V.I., Malina, O.V.: Method of creating a classifier of material objects class by the example of the transmission scheme with intersecting axes. In: *Problems of Characterization, Analysis and Logic Control, Academic Collection of Scientific Papers*, pp. 80–89. Moscow MGGU (1999)
4. Goldfarb, V.I., Malina, O.V.: Skew axis gearing scheme classifier building technique. In: *Proceedings of Tenth World Congress on the Theory of Machines and Mechanisms*, vol. 6, pp. 2227–2232, University of Oulu (1999)
5. Lozin, V.V.: Gearing optimization. *Optim. Eng.* **9**(2), 201–211 (2008)
6. Malina, O.V.: Features of using tree structures in CAD systems. In: *Proceedings of the Symposium Computer Mathematics in Informatiology*, pp. 8–11. Moscow-Izhevsk, IzhGTU (1997)
7. Malina, O.V.: Information models in problems of structural synthesis. *Sci. Tech. J. Comput. Math. Moscow Publishing Phys. Math. Lit.* **1**, 184–193 (2001)
8. Malina, O.V., Urzhumov, N.A.: Optimization of medium complexity structural synthesis process. *IzhGTU Reviewer* **1**(33), 144–150 (2007)
9. Malina, O.V., Zarifullina, E.G., Valeev, O.F.: The approach to construction of the engineering objects classifier as the basis of CAD information support. In: *Proceedings of Tula State University. Technical sciences, TGU*, vol 6–1, pp. 220–229 (2013)
10. Morozov, S.A., Malina, O.V.: Optimization of the synthesis process in automated systems of blank production preparation. In: *Proceedings of 10th International Symposium on Mechatronics*, pp. 269–273, Trencinske Teplice, Slovakia (2007)
11. Polovinkin, A.I.: *Theory of New Machinery Design: Regularity of Engineering and Its Using*, 140 p. Moscow: Informelectro (1991)
12. Wojnarowski, J., Kopeć J., Zawiślak, S.: Gears and graphs. *J. Theor. Appl. Mech.* **44**(1), 139–162 (2006)
13. Zanasi, R., Sandoni, G., Visconti, A.: Dynamic model and control of a gearbox system Available via Dipartimento di Ingegneria “Enzo Ferrari” http://www.dii.unimo.it/~zanasi/didattica/Veicolo_OLD/Twente_02_GearBox.pdf. Jan 2014

Part V
Miscellaneous

Search Module as a Tool for Improvement of Classifier

E.G. Zarifullina, O.V. Malina and I.M. Nekipelova

Abstract The article is dedicated to the approach of information search realization as part of a configuring system of engineering products. Effectiveness of a search depends on algorithm quality, trained on a database of the configuration system and base of linguistic rules, some of which are cited as examples in the article. Additionally, an algorithm is given for building a classifier and rules of data used for constructing the most optimal variants of search according to wishes of a user. From such a point of view, a search for information is a replenishment tool of the configurator database. It makes the system more open.

Keywords Search algorithm · Linguistic rules · Tree graph · Generalized model · Search engineering information · Optimization of search algorithm · Structural synthesis

1 Introduction

Today graph theory embraces many scientific problems and is being actively developed, because it provides a very effective way to present some complex systems such as nets. Specifically, it allows for formalize. Graph theory is still used to solve a large number of different tasks. In the process of engineering product development, one of the problems was formalization of complex objects with the use of graphs for correction and addition of the classifier of engineering products. A classifier is informational model of configurator created within the task of

E.G. Zarifullina (✉) · O.V. Malina · I.M. Nekipelova
M.T.Kalashnikov Izhevsk State Technical University, Izhevsk, Russia
e-mail: zarifullina@gmail.com

O.V. Malina
e-mail: malina_0705@mail.ru

I.M. Nekipelova
e-mail: irina.m.nekielova@mail.ru

structural synthesis. It is needed especially for forming of product structure, its analysis and modification [8].

Most universal automatized systems of engineering activities allow us to check the design selected by an engineer, to make calculations, to form technical documentation. However, the success of a product structure according to the design specification of an existing system cannot be guaranteed. This is mainly due to the fact that the development of a new or an existing product structure analysis is related to the creative nature of the designing process, which such a relation is difficult to be formalized. However, such systems can substantially facilitate the work of a designer due to a partial automation.

Today information systems quickly lose their relevance. They are associated with high-speed appearance of new (unprocessed) information and knowledge. The base needs regular replenishment of relative information from the domain of particular knowledge, because variants that impossible before, now have become acceptable. For example, development of new strong, lightweight materials has lead to the necessity of revision of production of many parts, which required an analysis of the old and adaptation of the new.

Searching for new information and applying it is a difficult but realizable task. Of course, as in any search, the search result may not be unambiguously reliable. However, the development of an optimal algorithm and introduction of it into practice can produce a significant automation of database replenishment by new data, attraction of new knowledge and its quick implementation, optimization of system tasks and creation of a broad platform for forming an experimental base, with minimum dependence on previous experience of the person in the object area.

2 Classifier as a Database of Search Module

In our work we use the previously developed classifier of gear systems as the database. In this formulation, the classifier is the information base for synthesis of medium and high complexity products (gearboxes) [3]. The heart of the classifier is a graph-table model of engineering complex objects proposed by Malina [2].

An engineering product classifier is based on the following algorithm:

1. Decomposition of the physical object (for example, a gearbox) breakdown in hierarchical order from complex to simple. The tree graph is the most successful implementation and presentation of complex systems (see Fig. 1).
2. Characterization of each graph element (node) that indicate the descriptive categories and values. Usually the object is described by many parameters, for example, the category of “mass” and the value “27,500 kg” or category “type of connection” with the value “soldering”. The characteristics as well as physical objects that may form a hierarchical structure, reflecting complex bounds among them.

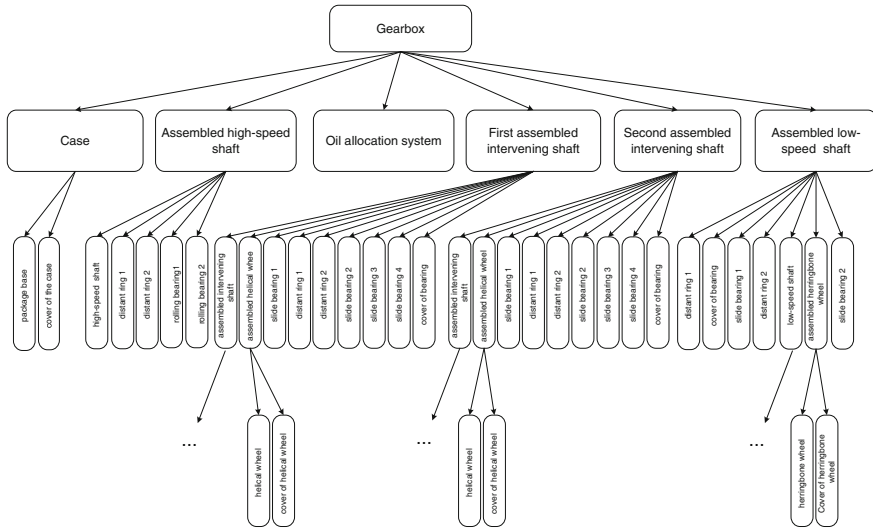


Fig. 1 Variant of a gearbox decomposition

3. Combine some decompositions into the generalized model that describes the class of objects united by a common functional task/item. Such a model is also presented by the tree graph in Fig. 2 that contains not only the functionality nodes, but types of connection AND-OR and features that combine multiple values within the meaning of different categories of decompositions. The generalized model is a tree-graph, where its root node is a complex object such as a gearbox. The tree-graph of a gearbox is presented in Fig. 2.

Creation of the classifier is presented in table form. Each row contains a single functional node of the graph is shown, columns contain characteristic values of the node. Filling columns depend on availability of the value in a given decomposition. A table method reveals emptiness in descriptions of the engineering object and allows store and supplement the class of objects model in the most optimal form.

Thus a classifier has information about the “all” class and is an information base that can share many processes such as synthesis, configuration of items, modification, etc. Relevance state of classifier and its replenishment are becoming obligatory conditions of quality designed, studied projected and serviced engineering products. So, the classifier is an open system, bounded with the environment with need of continuous information replenishment. “Open systems can exchange with surrounding objects by energy, matter and, which is also very essential, by information” [1]. There are several variants of retrieving information from the environment: firstly, directly by using expert knowledge, secondly, automated extraction by specialized tools of retrieving data from an information media such as the Internet. The main features of the expert’s work are point-direction and competence.

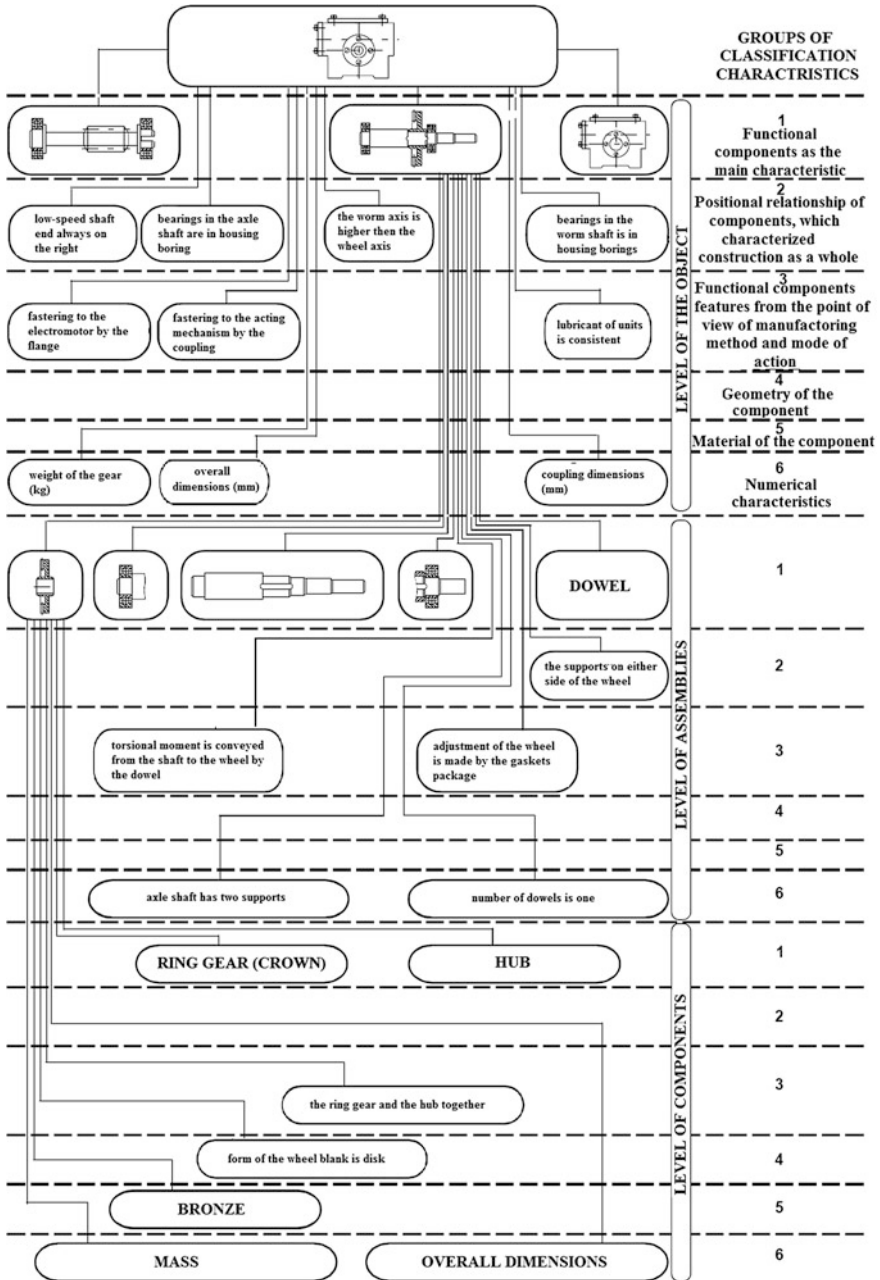


Fig. 2 Fragment of the tree-graph model after generalization step

When an expert adds information the requirement «a priori» is fulfilled. As a rule, the expert adds clearly defined and heuristically checked information. On the other hand, automated data collection tools focus on reduction of large array complex data, but the machine is not able to adequately assess the relevance and adequacy of the data. In any case, it needs expert help at the stage of selection.

The classifier may be resulted in an equilibrium state at the moment if database content indicators reach a defined level. Being in an equilibrium state the system is able to solve the tasks with a given degree of accuracy. The feature of extracting data from informational areas is a high degree of uncertainty of interaction results, associated with large entropy of the environment and “painful” ordering of an artificial system. An important feature of artificial systems is their high level of organization, thus initially they have equifinal state, i.e., a state defined only by internal structure of the system, that doesn’t depend on the state of the environment. Such systems evolve toward increasing level of complexity.

The increasing level of complexity is due to the addition and change of data into the classifier. Based on the graph model of data, such objects shall be replenished or excluded (for example, information about the loss of relevance of the data “connection elements by soldering”): functional nodes that are personification of a physical object, features and outcomes (values) that describe the object, types of connections (for example, earlier obligatory design elements can be optional by the results of the new products development) and forbidden variants of various elements of the class of objects, which also can be extracted from the classifier. The task is difficult, but realizable, because the language is the classification of high reliability [6]. All language operations are primarily based on logical operations: synthesis and analysis [5].

3 Information Search Algorithm

One of the main sources of knowledge is the Internet—one of the most difficult area in the processing of data and information, in general. Difficulty in extracting knowledge are: firstly, size and disarray of information, secondly, inadaptability to technical systems, except of certain strictly formalized information (technical guides, specifications), thirdly, a high rate of creation of new information, and fourthly, the need of realization algorithms of conversion information into data, because the second point is a formalized view of the first [7].

Types of queries by type of final result: complemented database; made more exact the available data; refinement, leading to the amendment; updating, leading to a change. In this work the search algorithm doesn’t allow for wittingly finding new, indeterminate information, loosely coupled with data from the classifier. For example, we know from the classifier that a gear box shifts rotational speed, but result of our search has shown the other possible object for shifting rotational speed. So, in this case we speak about point searching in desired results area. For instance,

search information about possibility of using concrete lubricant or finding a functional analogue of a failed unit.

As a result of analyzing the specific topics of pipe fittings major groups of topics are identified. They have an important meaning for work of the algorithm of search and selection of information.

- knowledge about technical and quality characteristics of existing elements of pipe fittings that allow us to make their comparisons and choose the best option for the calculations;
- knowledge about methods and peculiarities of using technologies: repair and change the device configuration, troubleshooting, etc.;
- knowledge about new devices, new principles of operations, description of tests, test reports, etc.

Knowledge may be represented in various forms:

- affirmation, for example, *“For the production of cast-iron can be used iron brands C4-15, C4-20 and antifriction cast-iron”*,
- facts, such as *“Geared production of King Right Motor are equipped with planetary, cylindrical or worm gear”*,
- conjectures, for example, *“Yes, G3 4" h21,8 seems exactly what you need”*,
- sum-total of the above, for example, *“Bought Blueweld 185. Detected that directly connection to 40-liter bottle of argon does not work, because the gearbox is designed for smaller thread diameter”*.

The task is to find the most relevant information, as well as to evaluate its contents by users request. Estimation of the content available via the formation of rules and query database, which includes a base of classifier, as carrier of accurate information. But for text analysis need specific categories, such as “assessment”, “actions”, etc. Thus, the main groups, fixed in the query database, includes knowledge:

Data from the classifier:

- names of structural elements and their functional description;
- brands, series, models, modifications of elements and products of pipe fittings;
- various features and descriptions of pipe fittings items (*promptness, speed, calculated traction power, convenience and others*);

Data, necessitated by the requirement to maximize of information extracting effectiveness from non-formalized information environment:

- names of manufacturers (SHAYANG YE INDUSTRIAL, King Right Motor, JSC “Reducer” et al.);
- the names of the companies, consumer organizations (*JSC “Nizhny Tagil Metallurgical Plant”, JSC “Evrazruda”, JSC “Tyazhmash”, Federal State Unitary Enterprise NPO Lavochkin, etc.*);
- estimately qualities of pipe fittings elements (*modern, cool, good, breaking, buggy, is unacceptable, and the like*);
- degree of probability (*possibly, probably, may be impossible, and so on*);

- emotional and evaluative characteristics by users (*like, use it, worn out, has fitted, set up, etc.*);
- area of using (*in production, engineering, for household use, etc.*);
- a list of actions (*used, produced, tested, etc.*);
- a list of common and individual abbreviations (*QF—quality control, TW—total weight, etc.*);
- possible variations of folksy (popular) element names.

Correctly implemented semantic rules identify and fix actual for users information, organizing communication between the groups presented above.

Example of text processing using introduced semantic rules:

1. Recognition of the text, highlighting key words.
2. Analysis of key words using semantic rules for example the keyword in the previous example, denotes the action “*is used*” Fig. 2.

As a rule, the key words are lexical units of action (verbs), the exact values (M12), names of items or parts (gearbox, cog-wheel), features and values of the functional tree-graph node of classifier (the number of stages—2). Taking into account such specific character we can create analysis information rules. Figure 3 shows a semantic network for keyword “used” (it is Participle II form in English and Participle in Russian). In this example logic was realized for English, however the main development works are carried out for a Russian and Russian-speaking Internet (Fig. 4).

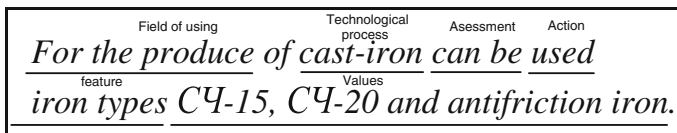


Fig. 3 Fragment of the text with analyzing logic

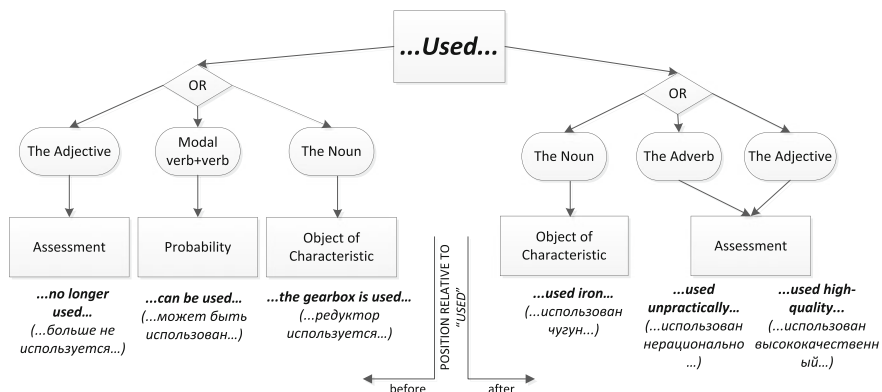


Fig. 4 Semantic rules of analysis an information by the example of English

Text analysis is optimized by the following actions:

1. Single-point query involving a narrow range of data from the classifier for the task of replenishment of the classifier. For example, request by mask “functional node + parent node + feature” like “*material gear wheel*”, firstly we get information about the possible values of the wheel attributes: *steel, capron*, etc.
2. Using the key groups of categories, depending on the desired result and tasks: estimated quality of the mechanism (e.g. query, *high quality gear*, you can get varieties that reveal the quality of gear), variants of the action (*running gear repaired*), degree of probability (*probably will work*) and etc.
3. Auto offers the best variants for query taking into account interesting for user the item from the classifier or desired action with it: search for alternatives, finding producers, calculations, addition functionality, etc.
4. Screenings known for classifier data if user want to find something new, such as finding alternatives, i.e. deleting of those alternatives that are marked in the classifier as an alternative to our required object. Alternatives are visualized in the generalization of several decompositions.
5. Selection of data, according to well-known for classifier, if user wants to confirm some information from the base of classifier.

4 Conclusion

Optimization mechanisms are very important, especially in language [4], so the system of configurator does not contradict the natural linguistic processes.

The search module is part of the under-development configurator of mechanical engineering produce at the base of Izhevsk State Technical University.

There are many effective search and attempts of text analysis algorithms (famous search engines Google, Bing, Yahoo, etc.). However, consideration of the search module as part of the complex system, that supports informational relevance, is making the research very perspective.

References

1. Klimontovich, Yu.L.: Entropy and information of open systems. Prog. Phys. Sci. **169**(4), 444 (1999)
2. Malina, O.V.: Features of using tree structures in CAD systems. In: Proceedings of the Symposium Computer Mathematics in Informatology, pp. 8–11. Moscow-Izhevsk, IzhGTU (1997)
3. Malina, O.V., Zarifullina, E.G., Valeev, O.F.: The approach to construction of the engineering objects classifier as the basis of CAD information support. In: Proceedings of Tula State University. Technical Sciences, vol. 6–1, pp. 220–229, TGU (2013)

4. Nekipelova, I.M.: Algorithm of equal search for the selection of linguistic resources in the speech optimizing process. *Mod. Res. Soc. Probl.* **4**(24) (2013). doi:<http://dx.doi.org/10.12731/2218-7405-2013-4-50>
5. Nekipelova, I.M.: Synthesis and analysis: ways of thinking and cognitive mechanisms of language activity: philology. *Theor. Pract.* **5**(23), II, 158–160 (2013)
6. Nekipelova, I.M., Zarifullina, E.G.: Language as natural system of multi-level classification of high reliability. *Mod. Stud. Soc. Probl.* **6**(26), 37(2013)
7. Zarifullina, E.G.: Language and speech: database and information: philology. *Theor. Pract.* **11**(41), 80–82 (2014)
8. Zarifullina, E.G., Malina, O.V.: Approach to building of information model of the configurator. *Intell. Syst. Prod.* **1**(23), 41–44 (2014)

Kazimierz Kuratowski—Biography and Genesis of the Theorem on Planar Graphs

J. Wojnarowski and S. Zawiślak

Abstract The present paper discusses Professor Kazimierz Kuratowski's achievements, especially proving his theorem on planar graphs in 1930. Some facts of the biography are analysed, aiming for explanations of how it was possible for him to do this and to delineate the background of such success. The general situation of mathematics in Poland (especially in Lvov) in the 1920s and 1930s is roughly described. The way of proving the theorem is also analysed i.e., transformation of the problem from the graph theory field into the field of topology.

Keywords Kuratowski theorem—multidisciplinary approach · Planar graphs · Lvov school of mathematics

1 Introduction

The Kuratowski theorem [1, 2] for planar graphs [3–8] is cited in almost every book on graph theory. The proof of this problem was published in 1930 [2], but the first formal announcement had been published in 1929 [1]. One can ask the question of how it would be possible in Poland—with rebirthing taking place in 1918 after World War I (WW1) after the lack of statehood for over 100 years. Moreover, in fact, this peaceful period started after the war against Soviet Russia in 1920. Additionally, there are no other results of Kuratowski's that were dedicated to graph theory.

J. Wojnarowski

Faculty of Mechanics and Technology, Silesian TU, 18a Konarskiego St., 44-100 Gliwice, Poland

e-mail: jozef.wojnarowski@polsl.pl

S. Zawiślak (✉)

Faculty of Mechanical Engineering and Computer Science, University of Bielsko-Biala, 2 Willowa St., 43-309 Bielsko-Biala, Poland

e-mail: szawislak@ath.bielsko.pl

© Springer International Publishing Switzerland 2017

S. Zawiślak and J. Rysiński (eds.), *Graph-Based Modelling in Engineering, Mechanisms and Machine Science* 42, DOI 10.1007/978-3-319-39020-8_18

233

There are several papers and books on Kuratowski's biography and scientific achievements. As for data connected with his life, the most important is his autobiography [9] followed by a number of historical books and papers [3, 4, 10–34].

The problem of planarity for graphs can be formulated in the following manner: to draw a graph on a plane in such a way that the edges have common points only in graph vertices so there are not any crossings of the graph edges. It is not clear when the problem was firstly formulated, whereas a historical approach to graph drawing (in general) was presented in a very interesting way in [26]. According to the thesis [17], the problem of planarity was mentioned e.g. by Moebius in 1840 and Dudeney in 1917 (also the bipartite $K_{3,3}$ graph problem was formulated). According to the thesis [17], the planarity problem is connected strictly with graph coloring problems (such as the 4-color theorem), and the graph duality problem as well as the study of polyhedra.

The goal of this paper is to outline a hypothetical explanation of how it was possible for Kuratowski to solve this problem. According to our best knowledge, the genesis of his proof of the problem was not point-blank analyzed despite the fact that several historical papers, theses and book had been published. The only existing paper emphasizing that the proof is not done in terms of graphs is [10], according to the best knowledge of the Authors of the present paper.

There are papers dedicated to the proof of the planarity problem [3, 4, 10, 12, 35–37] but sometimes they focus an attention on the question: ‘was Kuratowski really the first to do this?’ [12, 22]. The paper that traces versatile attempts of proving the planarity theorem is paper [22]. The authors described the fact that also American scientists Frink and Smith [15] did the proof but their paper was rejected because the Editors had knowledge about Kuratowski's publication. However, the announcement was even later published than the complete proof of Kuratowski, therefore their full work dedicated to this problem was finally not printed. There are also publications named the theorem on planarity such as the Kuratowski-Pontrjagin theorem [12] (mainly in Russian publications) but the paper [22] explains the history in details based upon relations of Kuratowski's co-workers i.e., only oral communications were exchanged between scientists from Poland and Russia. The early Polish version of the contribution on the transformed problem of planarity, translated recently into English, can be found in [38].

There is no separate book on Kuratowski's education path or his scientific curriculum vitae and achievements such as exist for other famous Polish mathematicians of this era e.g., Stefan Banach [19–21, 24, 31, 32]. However in book [11], there is a chapter written by R. Engelking entitled “Kazimierz Kuratowski (1896–1980) His life and work in topology”—pages: 431–452. There are several short papers [16, 22] dedicated to the biographical data. However, the books and the papers do not fully analyze the cases of this outstanding success according to planar graph problems.

Moreover, in the present paper, the historical background will be studied. This background is embedded in the history of creation of the so-called ‘Lvov school of mathematics’ [13], Warsaw school of mathematics or finally Polish school of mathematics within the period 1918–1939. In that time, the internet, on-line

electronic versions of scientific journals and even television were not available. Some facts from earlier history of Poland are also shortly mentioned.

The considerations are partly based on memoirs of Kuratowski himself [9], but also Hugo Steinhaus [33], Kazimierz Szałajko [34] and Stanisław Ulam [39].

1.1 General Historical Background

Poland was erased from the map of Europe in 1795. It was conquered by neighboring countries Russia, Prussia and Austria. In those times, the territory of Poland enclosed some lands of the current Lithuania, Belarus, Ukraine and partly Latvia. It was due to peaceful treaties with Lithuania—so called Unions (years—1385 (Krewo), 1413 (Horodło), 1501 (Mielnik) and 1569 (Lublin)). Poland was created anew in 1918. The borders of the country were established during the Versal (Paris) conference, the uprisings (Silesia and Poznan regions), local voting (plebiscites) as well as—finally—the war with Soviet Russia (1920) and the post-war peace treaty. The borders, which were finally established, encircled Vilnius and Lvov as Polish towns.

During the time of division, Poles were citizens of the states of invaders. They had to learn the appropriate state languages. Simultaneously they could study in the best Universities of these countries.

2 Mathematics in New Established Poland

The countries that conquered Poland performed a policy of assimilation of citizens—consisting in: introducing state languages in the school system from grammar schools up to universities, in courts as well as in the state offices and institutions. At the beginning, the job carriers of Poles in Russia, Prussia and Austria were rather rare in such domains as politics, army, kingdom or empire officers (courts, authorities) or science. However, some Poles were active in France, Switzerland or Great Britain and other European countries. However, within early years of the XX-th century, the Austro-Hungarian Empire changed the internal national policy, smaller changes took place also in Russia and Prussia. For example, the Universities in Austrian province ‘Galicia’ e.g. Lvov and Cracow could give lectures in Polish.

In 1918, in the new-established Poland, 5 universities were created: Warsaw, Cracow, Poznan, Lvov and Vilnius, moreover some technical universities were also established in Warsaw, Cracow, Lvov and a lower rank technical school in Poznan. The university in Warsaw started its works in 1915, just during the war, not waiting for formal establishing of an independent country. It caused that the curricula or/and contents of lectures were not supervised by government, ministry or even based on inter-university exchange.

The basics for such immediate and efficient opening of the universities are versatile activities performed at the end of XIX century within the former territory of Poland:

- foundation of societies: Cracow Academy of Science in Cracow (Austro-Hungary Empire), Society of Science in Warsaw (Russia),
- foundation of journals (Mathematical and Physical Works, Bulletin of Academy Skills in Cracow, Mathematical Newsletter),
- opening of Scientific Courses (Warsaw, Russia), introducing Polish language in universities in Austro-Hungarian Empire.

The new Polish Universities and Technical Universities hired as professors or lecturers many outstanding mathematicians:

- (a) returning from other Universities after studies or doctorate. Here only a few exemplary names are listed:
 - Kazimierz Żorawski (pupil of Sophus Lie from Norway)—differential geometry firstly in Cracow, then in Warsaw (from 1919),
 - Stanisław Zareba—study in Petersburg (engineering), University of Paris (mathematics), doctor thesis at the Sorbonne University in Paris,
 - Zygmunt Janiszewski—doctor thesis in France, doctor exam commission consisted of: H. Poincare, H. Lebesgue and M. Frechet,
 - studying abroad: Stefan Mazurkiewicz (Munich, Göttingen), Hugo Steinhaus, Waclaw Sierpiński (Göttingen);
- (b) returning to Poland being professors in other countries:
 - Antoni Przeborski—Charkov to Warsaw,
 - Jan Sleszyński—Odessa to Cracow,
 - Wiktor Emeryk Staniewicz—Petersburg to Vilnius,
 - Leon Jan Staniewicz—Petersburg to Warsaw (after WW2 in Gdansk).

There were also other professors (who returned) from other fields of knowledge related to mathematics e.g.:

- physics: Franciszek Zienkowski (Göttingen to TU Warsaw) who at the beginning was a Kuratowski's teacher in the secondary school in Warsaw [9]; Wojciech Rubinowicz (Muenchen to Lvov) co-worker of Arnold Sommerfeld [13, 18–20, 23], Józef Wierusz-Kowalski (Switzerland to Warsaw University);
- astronomy: Marcin Ernst [23] (diplomas: Berlin and Vienna—work at the Lvov University).

There were also a reverse directions of migration of scientists in 1930s i.e. newly educated Polish young graduates or doctors emigrated to USA or Europe (especially of Jewish origins) e.g.: Stanisław Ulam (Lvov—Los Alamos, Princeton, USA), Otto Nikodym (Lvov—Kenyon College, USA), Marek Kac (USA), Leopold Infeld (Cambridge, U.K., Princeton, USA). However, they periodically returned to Lvov for seminars or during summer holidays. In every case, meetings with all

former co-workers were a must. It allowed for immediate exchange of ideas and achievements.

Alfred Tarski, mathematician from Warsaw University, emigrated to USA in 1939, working e.g. in Harvard University, Princeton and Berkeley.

It can be stated that mathematics in Poland in this time was on a very high level which can be confirmed by the possibility of working of Polish mathematicians in the best world-wide universities as well as working on problems of extreme novelty like e.g. Stanisław Ulam worked on the atom bomb in Los Alamos (being educated and doctor promoted in Lvov).

Moreover, many outstanding mathematicians from all-over the world visited Poland especially the Universities in Lvov and Warsaw (see Chaps. 2 and 3).

2.1 Methods of Working and the Scottish Book

The methods of co-operations between the scientists of this period in Poland were amazing and rather outstanding. They established an informal [39] collective, groups of fellows. They called themselves “colleague”, they discussed problems in different cross-domain and multi-university groups—for example the informal meeting of the Polish Mathematical Society took place regularly almost every Saturday evening, besides the seminars organized during working days at the Universities [39, p. 67]. Cooperation of the Lvov branch with branches in Cracow, Poznan or Vilnius was live and frequent. The report on formal meetings in the Lvov branch covering the period 1928–1938 [13] had shown that there were 180 meetings and 360 talks which means that there were on average 16–17 official meetings per year and usually two talks were given in every case. In total, there were 65 authors including 16 authors from abroad.

Many famous mathematicians from other countries were registered as foreign members of the Society. The presidents of the Society were consecutively chosen from different Polish Universities (Stanisław Zaremba-Cracov, Wiktor Staniewicz-Vilnius, Zdzisław Krygowski-Poznan, Waclaw Sierpiński-Warsaw, Kazimierz Bartel-Lvov) [21, 27, 40].

The state congresses of the Polish Mathematical Society started to be organized in 1927. The first took place in Lvov (J. von Neumann (USA), M. Luzin and N. Bari (Russia), V. Hlavaty (Prague) were the guests), the following in 1931 (Vilnius) and in 1937.

The informal meetings of mathematicians took place in different places but one of Lvov’s restaurants was very characteristic and attended regularly. It was the Scottish Café placed in Fredro square. The meetings among others consisted in posing and discussing mathematical problems. If the problem was solved, those who discussed designated one who wrote down the results, all real contributors who really developed the considerations were automatically the authors of the immediately prepared paper. Since 1933 (or 1934 [39] or 1935 [13]) the problems had been registered in a special copybook which was called the “Scottish Book”. The register (from a copied original version) was translated into English and edited by S.

Ulam [41] in USA. One of the authors of the present paper had the privilege of reading the original. Kazimierz Szalajko, a co-worker of Stefan Banach, was a director of the Institute of Mathematics at the Silesian University of Technology in Gliwice in 1970s. Upon his invitation, son of Stefan Banach came for a meeting at the Faculty of Applied Mathematics, showing the original copybook which was preserved so many years also during the World War II (WW2). The xerox copy of the book is stored in the Library of the Polish Academy of Science Institute of Mathematics in Warsaw. The problems were solved by different mathematicians, sometimes after several years later.

The papers of Polish mathematicians were published sometimes in Polish but mainly in foreign languages. Moreover some papers were published not only in the created Polish journals but also in the best world-wide journals, e.g., Bulletin of the American Mathematical Society [39, p. 71].

2.2 *Journals*

Immediately after creation of independent Poland, Polish scientific journals of international meaning were established [9, 13, 27, 32] i.e.: “Fundamenta Mathematicae” in 1920 in Warsaw and a little later “Studia Mathematica” in Lvov in 1929 as well as “Acta Arithmetica” in Warsaw in 1935. The advantages of these journals were as follows: the published works were of very high scientific level, they were published only in foreign languages (French, German and English), 18 % of papers were published by authors from abroad [13]. The additional benefit was that almost 200 items of one issue (500 items of the first mentioned title) were subjected to exchange with other world-wide journals and university libraries [13, p. 56–58] due to lack of financial sources just for buying. A result was that not only Polish achievements were distributed around the Europe but also Polish scientists could have access to the newest achievements across Europe. The editors of the first issue of “Studia Mathematica” were: Stefan Banach and Hugo Steinhaus.

2.3 *Congresses, Conferences and Visits*

In the discussed times i.e. 1920s and 1930s, there were no such social media as the internet (e.g.: ‘Scholar Google’, SpringerLink data bases) or mobile phones, therefore live meetings and establishing of personal connections between scientists had been a base of knowledge about the state-of-art in a particular field of science. Circulation of journals was not so fast as today’s on-line journals which causes that nowadays availability is immediate. Participation in congresses, conferences and

lectures was an effective way to be informed on the current trends and achievements. Moreover, it is well worth emphasizing that Polish mathematicians were not only participants in those events but also invited speakers. It was due to knowledge and appreciation of their contribution to particular fields of science. There were several lectures given by Poles in some famous and prestigious universities e.g.:

- (i) Kazimierz Kuratowski: 1927—invited lectures Heidelberg and Goettingen (meetings with P. Aleksandrov, H. Hopf, E. Noether, D. Hilbert), 1935—Princeton, Harvard, USA,
- (ii) Stanisław Ulam: 1934—visit in Vienna—Professor K. Menger (topology, geometry) formerly invited to Poland by Kazimierz Kuratowski, 1934—visit in Paris—Professor E. Cartan, 1934—visit in Cambridge, U.K.—Professor G.H. Hardy.

Visits of foreign mathematicians in Lvov were relatively frequent. Some famous visitors were as follows: Austria (M. Jacob.), France (E. Borel, J. Leray, P. Montel, H. Lebesgue), Czech Republic (E. Čech, W. Hlavaty), U.K. (G.T. Whyburne, A. C. Offord, A.J. Ward), USA (J. von Neuman), Germany (E. Zermelo, L. Lichtenstein), Swiss (R. Wawre), Romania (P. Segrescu, S. Stoimov), Russia (N. Bari, N.N. Bogolubov, L.A. Lusternik, N. Luzin, S.L. Sobolev) [3, 17, 23, 26]. The title of Doctor Honoris Causa of Jan Kazimierz, University of Lvov was granted to the famous French mathematician Henri Lebesgue in May 1938.

Polish scientists took part in three Mathematical Congresses which were held in: (a) Bologna (1928), (b) Zurich (1932) and (c) Oslo (1936). Professor Kuratowski took part in all of these congresses. In the first of them Hugo Steinhaus and Stefan Kaczmarz had contributions in the conference sections. In Zurich, Professor Waclaw Sierpiński gave a plenary lecture. In total, there were 20 Polish participants [13]. In Oslo, the plenary lecture was given by Stefan Banach. Other congresses in which Poles took part were as follows: 1930—Charkov Mathematical Meeting (Antoni Przeborski, Jerzy Sława-Neyman) [27], 1934—Conference on Differential Geometry in Moscov (Antoni Hoborski, Stanisław Gołąb, Aleksander Wundheiler), 1935—Conference on Topology in Moscov (Karol Borsuk, Witold Hurewicz, Stefan Mazurkiewicz, etc.) [27].

3 Short Kuratowski's Biography (1896–1980)

The facts about Kazimierz Kuratowski are collected mainly from his autobiography. He starts the book with a statement on the first page (!) that He began to learn foreign languages from early childhood [9, 42, 43]. In fact, French was taught simultaneously with Polish as a second language at home. At school English and German were taught. Living in the Russian Empire, Kuratowski had to speak Russian which was used at schools and in administration. To attend and finish

school, Russian should be known fluently. Then, throughout his life, Kuratowski continued to study and consequently spoke all these 5 languages perfectly.

Parents as well as their relatives were lawyers, medical doctors, historians, engineers etc. They were well educated and rich. His father was also a journalist in the local newspaper for lawyers, moreover he was involved in versatile social organizations. Brother Roman was a lawyer. Kuratowski was born in Warsaw and spent his youth in this town. He attended schools of an extremely high level of education. In 1913 he started to study at the Glasgow University at the Engineering Faculty. It was popular that rich citizens of Polish nationality sent daughters and sons to study abroad but Glasgow was not too popular. Kuratowski's choice was supported by a will to go with one of his best colleagues. He finished first year with the second rank and returned for holiday to his parents. He was satisfied by the lectures of Andrew Gray formerly assistant to Lord Kelvin. Additionally, in Glasgow mathematical lectures were common in the first year for engineers and mathematicians—so the level was really high. In 1914 World War I began and after the war he did not return to Glasgow but started to attend University in Warsaw which was opened in 1915. It was during the war therefore that the study curricula were not obeyed and lectures were arranged according to availability of professors. A result was that he was unbalanced on the side of geometry having all of its kinds: analytical, projective, differential, descriptive etc. He also gave lectures on topology which was a new branch of mathematics (Zygmunt Janiszewski, Waclaw Sierpiński). Kuratowski's first scientific results were achieved during his studies therefore publishing his work at the University was an obvious option. He had published several papers as a student and additionally other results were randomly collected, so the problem was only in formal paper writing. He obtained a doctoral diploma (PhD) on the 5th January 1921. In the Fall, Kuratowski obtained the title of D.Sc. (habilitation) on set theory and started to work as an associate professor. Within the period 1917–1926, he published approximately 40 scientific papers of great importance and therefore obtained a proposal to move to Lvov. Kuratowski accepted the opportunity. In 1927, He traveled to Heidelberg and Göttingen in Germany for invited lectures. Work at the Lvov University lasted within the period 1927–1933. Among Kuratowski's co-workers were e.g. Edward Otto (who was involved in descriptive geometry) and Stanisław Ulam. The Chair of Descriptive Geometry was directed by Stanisław Bartel. Kuratowski published many new papers there e.g. common papers with Stanisław Ulam on the Hausdorff measure problem and Edward Otto on topology. But the most famous is the graph planarity theorem published in 1930.

Kuratowski returned to the University of Warsaw where he worked till the end of his work duties. During World War II the Polish universities were closed but he took part in the so-called underground system of higher education which could be punished by death. After World War II, Kuratowski flourished in both areas: science and organization work. He was an active fellow of the Polish Mathematical Society as well as a founder and a member of the Polish Academy of Science. He had also world-wide recognition, giving lectures in the USA (1948/1949 and 1968/1969) in Harvard, Princeton, Yale, Columbia, Berkeley and Baton Rouge. Other destinations of his scientific visits were: China, India (1956), Germany (1957) and Russia.

Kuratowski received honorary degrees from many universities including Glasgow (U.K), the Sorbonne (Paris, France), Prague (Czech Republic) and Wrocław (Poland).

4 Proof of Kuratowski's Theorem—Genesis

It seems that the circumstances allowing for the successful solving of planarity problem were as follows:

(i) personal:

- gifted, from the family of lawyers, doctors, teachers, well educated,
- perfect knowledge of foreign languages: Russian—certificate of secondary school in Moscow, English—one academic year of study at Glasgow University (1913–1914) and lectures in USA, French—many publications, known from childhood, German—several visits in Germany, which allowed for reading, publishing scientific papers from versatile journals, conferences' proceedings as well as giving and hearing conference talks,
- probably the most important—continuation of studies in Warsaw since 1915—during the World War I based on a study curricula arranged to the available professors. It was a non-classic and a non-traditional study course. Therefore he studied at first several subjects connected with geometry in extended hours—especially topology and even descriptive geometry,
- mobility, many scientific trips abroad, attendance at mathematical international conferences and meetings, being informed on the current state-of-art,
- a period of work at Lvov University, descriptive geometry specialization of his doctoral student Edward Otto,
- lectures abroad e.g. in Germany (1927) and in the USA;

(ii) scientific and social:

1. rebirth of Poland—the majority of professors came to the newly-opened Universities from Austria, Russia and Prussia (outstanding and very high-level best Universities, doctors came also from France, Germany and Switzerland—obtaining PhD titles from the best European (or world-wide) professors,
2. high, international level of lectures and knowledge of current trends,
3. creation of three Polish mathematical journals of international meaning and via purchasing and also exchange—availability of foreign journals,
4. international co-operation on the highest level, the best world-wide mathematicians in Lvov, Polish mathematicians visited leading Universities in USA and Europe,

5. taught by outstanding professor and teacher—so-called personalities—which is in sound opposition to modern tendencies of distance learning. It seems that meetings with such supervisors who inspire was a trigger of his own will of self-improvement,
6. active work in Mathematical Society, organizing of visits of foreign professors, an active part in European Mathematical Meetings, which gave additional opportunity for exchange ideas,
7. contacts with Hugo Steinhaus [39, p. 72] who frequently proposed unusual approaches consisting in representation of geometrical problems via combinatorial approaches—i.e. reverse attitude from Kuratowski’s way. However, nowadays, it is worth underlining the need of posing reverse problems. It could allow for seeing both of them in a new light or from different points of view. Other multi-disciplinary approaches were considered by S. Ulam and J. Schreier—merging problems from topology and algebra.

The proof of the graph planarity theorem was published in 1930 [2] in French. However, some introductory results were published even earlier in Polish which

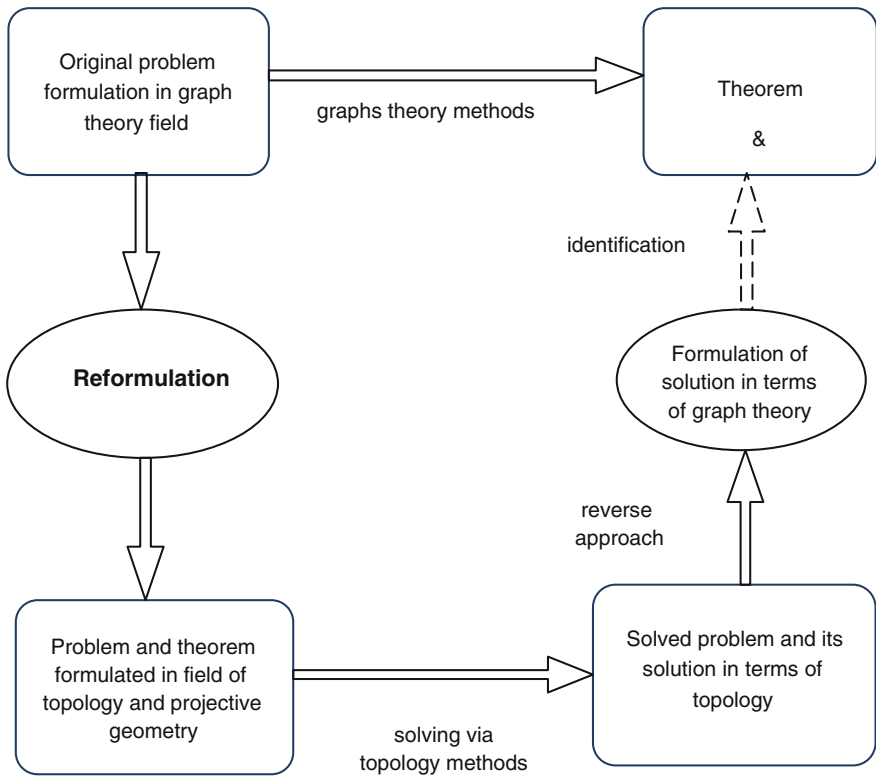


Fig. 1 Multi-disciplinary approach to solving the problems via their cross-field transformation upon planar graph problem

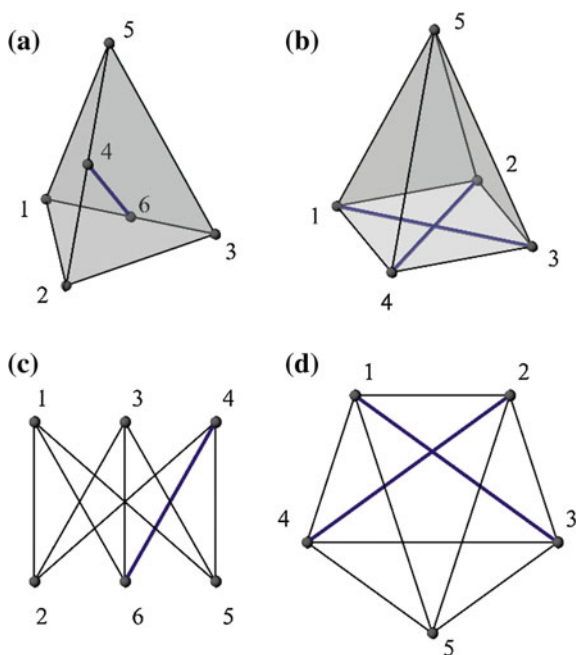
was described in historical publications [44, 45]. The theorem says that a graph is planar if it does not contain as subgraphs any graphs equivalent to K_5 and $K_{3,3}$. It had been proved in fact as a problem of topology or even projective geometry [10] (!) as it is mentioned in the French title of the paper which is available e.g. from webpage [42]. The combinatorial proofs of the theorem can be found in [29, 30].

The utilized approach can be symbolically presented in the scheme shown in Fig. 1. Based upon the original text of [2] one can see that the scheme presented in Fig. 1 was utilized by Kuratowski. For example: nowadays called Kuratowski's graphs: K_5 and $K_{3,3}$ (Fig. 2c, d) had been converted into two adequate polyhedra—pictures 1 and 2 [2, p. 272] which are redrawn in the present paper in the first row in Fig. 2. The numbers of polyhedral nodes are assigned to adequate graphs' vertices presented in the second row in Fig. 2c, d. In case of the first polyhedral, one edge was added to the first skeleton (Fig. 2a), i.e. (4, 6) and diagonals (1, 3) and (2, 4) were added in the second case (Fig. 2b).

As it was previously mentioned, there are no papers of Kuratowski dedicated to graph theory, written in the nomenclature of graph theory, which we know checking his complete bibliography [43].

In fact, the multi-disciplinary approach can be utilized only by scientists who are open-minded, have a wide scope of interest, and are eager to acquire novelty and originality. This idea has been propagated recently but to our surprise there are still opponents fighting for purity of some fields of knowledge. We do hope that the above considerations confirm the necessity of a wider and more interdisciplinary attitude to scientific problems, especially graph theory and its applications.

Fig. 2 Two polyhedrals originally considered by Kuratowski in [2] (first row, (a)(b)) and two adequate graphs—known as Kuratowski's graphs (second row (c)(d)) where the numbers of nodes are equal to the equivalent graphs' vertices names



Further investigation into this problem consisted in: a planar graph drawing by means of straight lines only [14, 46], a planar graph drawing in grids [47], algorithms for checking planarity [4, 48], some generalization on infinite graphs [49] and dual form of the discussed theorem [35]. Recent papers, where comparisons of planarity checking algorithms can be found, are [4, 50], even of polynomial complexity.

5 Conclusions

The paper discusses Kuratowski's theorem on planar graphs. The origins are analyzed taking into account the fact that Kazimierz Kuratowski was born in the period when Poland was not present on the map of Europe. The state was re-established in 1918 and almost immediately many outstanding Polish mathematicians were actively working in Poland and publishing works on a world-wide level. In the case of Kuratowski, he had been a professor of mathematics in Warsaw and Lvov since 1920s. The present considerations enclose the hypothetical genesis of the successes of the Polish school of mathematics and Kuratowski himself.

It seems that paradoxically Kuratowski's study during World War I could have been the base for his interest in planar graphs because he was taught in an imbalanced and untypical way on topology and geometry due to availability of adequate professors just at this time, additionally spending his first year at the Glasgow University.

Moreover, during partitioning of Poland citizens had to know foreign languages and Poles had to view their scientific carriers within temporary home countries (Russia, Prussia or Austria) or wherever in Europe and even in the USA. Many professors, due to their patriotic backgrounds, started to work for an independent Poland since 1918 after rebirthing of the state. It is worth emphasizing that Polish mathematicians organized their professional lives and organizations in outstanding efficiency, scope and manner. It gave the background for achieving a high international level and in some fields allowed even for world-wide leadership.

References

1. Kuratowski, K.: Sur les courbes gauches. *Annales Polonici Mathematici* **8**, 324 (1929)
2. Kuratowski, K.: Sur le probleme des courbes gauches en topologie. *Fund. Math.* **15**, 271–283 (1930)
3. Dirac, G.A., Schuster, S.: A theorem of Kuratowski. *Indagationes Mathematicae* **16**, 343–348 (1954)
4. Liebers, A.: Planarizing graphs—a survey and annotated biography. *J. Graph Theory* **5**, 1–74 (2001)
5. Whitney, H.: Planar graphs. *Fund. Math.* **21**, 73–84 (1933)
6. Wilson, R.J.: Graph theory. History of topology. Elsevier, Amsterdam (1999)

7. Wilson, R.J., Watkins, J.J.: *Graphs: An introductory approach*. Wiley, New York (1990)
8. http://en.wikipedia.org/wiki/Planar_graph
9. Kuratowski, K.: *Auto-biography* (in Polish). Czytelnik, Warszawa (1981)
10. Archdeacon, D.: A Kuratowski theorem for projective plane. *J. Graph Theory* **5**, 243–246 (1981)
11. Aull, C.E., Lowen, R. (eds.): *Handbook of the history of general topology*. Kluwer Academic Publishers, Dordrecht (2001)
12. Burstein, M.: Kuratowski-Pontrjagin theorem on planar graphs. *J. Comb. Theory, B* **24**(2), 228–232 (1978)
13. Duda, R.: *Lvov School of Mathematics* (in Polish). WUW, Wrocław (2007)
14. Fary, I.: On straight-line representation of planar graphs. *Acta Sci. Math. (Szeged)* **11**, 229–233 (1948)
15. Frink, O., Smith, P.A.: Irreducible non-planar graphs. *Bull. Am. Math. Soc.* **36**, 214 (1930)
16. Harary, F.: Homage to the memory of Kazimierz Kuratowski. *J. Graph Theory* **5**, 217–219 (1981)
17. Hudson, R.A.: *Planar graphs: a historical perspective*, Master-thesis, Department of Mathematics, University of Louisville, Louisville, Kentucky (2004)
18. Hunderson, D.: Banach’s space: Lvov and the Scottish Café. *Condens. Matter Phys.* **7**(4), 679–682 (2004)
19. Jakimowicz, E., Miranowicz, A. (eds.): *Stefan Banach. Incredible life and genial mathematics* (in Polish), Impuls, Kraków (2010)
20. Jakimowicz, E., Miranowicz, A.: *Stefan Banach: Remarkable life, Brilliant Mathematics*, Gdańsk University Press/American Mathematical Society, 185 p. (2012)
21. Kałuża, R.: *Stefan Banach* (in Polish). Wydawnictwo GZ, Warszawa (1992)
22. Kennedy, J.W., Quintas, L.V., Sysło, M.M.: The theorem on planar graphs. *Historia Mathematica* **12**(4), 356–368 (1985)
23. Krasinkiewicz, J.: A note on the work and life of Kazimierz Kuratowski. *J. Graph Theory* **5**, 221–223 (1981)
24. Koziński, J.: *Banach–genial scientists from Lvov* (in Polish). Wydawnictwo Akademickie ZAK, Warszawa (1999)
25. Kowalski, J.: Polish mathematical society. *Eur. Math. Soc. Newsl.* **54**, 24–29 (2004)
26. Kruja, E. et al.: A short note on the history of graph drawing. In: *Proceedings of Graph Drawing*, Springer LNCS, vol. 2265, pp. 272–286 (2001)
27. Kuratowski, K.: *A half century of polish mathematics: Remembrances and reflections*. (Translated by Andrzej Kirkor, Preface by Stanisław Ulam), vol. 1, Oxford, Pergamon Press (1980)
28. Kuzawa, M.G.: *Modern Mathematics: The Genesis of a School in Poland*. College and University Press, New Haven (1968)
29. MacLane, S.: A combinatorial condition for planar graphs. *Fund. Math.* **28**, 22–32 (1937)
30. Makarychev, Y.: A short proof of Kuratowski’s graph planarity criterion. *J. Graph Theory* **25**, 129–131 (1997)
31. Mauldin, R.D. (ed.): *The Scottish Book. Mathematics from the Scottish Café*, Birkhäuser, Boston (1981)
32. Przeniosło, M.: International mathematical journals published in Poland between the Wars. *BSHM Bull.* **24**(1), 20–26 (2009)
33. Steinhaus, H.: *Memoires and notes* (in Polish). Aneks Publishers, London (1992)
34. Szałajko, K.: *Memoires on Stefan Banach, remarks in the light of the Lvov city itself and the Lvov school of mathematics* (in Polish). *Opuscula Mathematica* **13**, 45–54 (1993)
35. Harary, F., Tutte, W.T.: A dual form of Kuratowski’s theorem. *Bull. Am. Math. Soc.* **71**, 168 (1965)
36. Thomassen, C.: Kuratowski’s theorem. *J. Graph Theory* **5**, 225–241 (1981)
37. Tverberg, H.: A proof of Kuratowski’s theorem. *Ann. Discret. Math.* **41**, 417–419 (1988)
38. Kuratowski, K.: On the problem of skew curves in topology, (translated by J. Jaworski) in Borowiecki, W. Kennedy, J.W., and Sysło, M.M. *Graph theory: proceedings of a conference*

- held in Łagów, Poland, February 10–13, 1981. *Lecture Notes in Mathematics*, No. 1018. Springer, pp. 1–13 (1983)
39. Ulam, S.M.: *Adventures of a mathematician* (in Polish), Prószyński i S-ka, Warszawa (1996); first published by University of California Press, Berkeley (1976)
 40. <http://www.ptm.org.pl/zawartosc/polskie-towarzystwo-matematyczne-w-latach-1919-1963>
 41. Ulam, S.M. (ed.): *The Scottish Book: A Collection of Problems*. Los Alamos Scientific Laboratory, Los Alamos (1957)
 42. <http://www-history.mcs.st-and.ac.uk/Biographies/Kuratowski.html> (in Polish)
 43. www.leksykon.ptm.mimuw.edu.pl/biogramy/Kuratowski/pliki/Kazimierz-Kuratowski-publicacje.pdf (in Polish) (webpages were opened on 29 November 2014)
 44. Borowiecki, M., Kennedy, J.W. and Sysło, M. (eds.): *Graph Theory*, Łagów 1981. In: *Proceedings of the Conference*, *Lecture Notes in Mathematics*, vol. 1018, Springer, Berlin, Heidelberg (1983)
 45. Kuratowski, K.: On the problem of skew curves in topology (translation into English by J. Jaworski), published in [44], 1–13 (1983)
 46. Wagner, K.: Über eine Eigenschaft der ebenen Komplexe. *Math. Ann.* **114**, 570–590 (1937)
 47. Schnyder, W.: Embedding planar graphs on the grid. *Proc. of 1st ACM/SIAM Symposium on Discrete Algorithms (SoDA)*, pp. 138–148 (1990)
 48. Kelmans, A.: A new planarity criterion for 3-connected graphs. *J. Graph Theory* **5**, 259–267 (1981)
 49. Thomassen, C.: Planarity and duality of finite and infinite graphs. *J. Comb. Theory, B* **29**, 244–271 (1980)
 50. Chimani, M., Zeranski, R.: Upward Planarity Testing in Practice: SAT Formulations and Comparative Study. *J. Exp. Algorithmics (JEA)* **20**, 1–2 (2015)

Author Index

B

Banerjee, J., 107
Barreto, R.L.P., 117
Borowik, B., 131

C

Cieśla, M., 51
Coaccioli, P., 97

D

Deptuła, A., 189, 201
Drewniak, J., 81, 131
Dubchak, L., 39

G

Garlicka, P., 131
Gola, B., 69

I

Ibănescu, R., 3

K

Karpinski, M., 39
Karpinskiy, V., 39
Kischnick, S., 25
Kopeć, J., 69, 81
Kutschenreiter-Praszkiewicz, I., 165

L

Lanzotti, A., 175

M

Malina, O.V., 211, 223
Martins, D., 117
McPhee, J., 107

Moreno, G.G., 117
Mrówczyńska, B., 51

N

Nekipelova, I.M., 223
Nicolazzi, L., 117

P

Partyka, M.A., 201
Patalano, S., 175
Pennestri, E., 97

R

Rysinski, J., 69

S

Shaikhanova, A., 39
Stryczek, R., 143

T

Tittmann, P., 25

V

Vieira, R.S., 117
Vitolo, F., 175

W

Wojnarowski, J., 233

Z

Zarifullina, E.G., 211, 223
Zawiślak, S., 69, 81, 233
Zolotov, A., 39