

**Studia i Materiały  
Informatyki  
Stosowanej**

# Studia i Materiały Informatyki Stosowanej

czasopismo młodych pracowników naukowych, doktorantów i studentów

patronat: Polskie Towarzystwo Informatyczne



POLSKIE TOWARZYSTWO INFORMATYCZNE

## Przewodniczący Rady Naukowej

prof. dr hab. inż. czł. koresp. PAN Janusz Kacprzyk, IBS PAN

## Redaktorzy Naczelni

dr inż. Jacek Czerniak, UKW

dr inż. Marek Macko, UKW

## Sekretarz redakcji

dr Iwona Filipowicz, UKW

## Komitet Redakcyjny

dr inż. Mariusz Dramski, UKW

dr Włodzimierz Masierak, UKW

dr Piotr Prokopowicz, UKW

dr inż. Grzegorz Zych, UKW

## Rada Naukowa

	dr hab. Stanisław	Ambroszkiewicz	Instytut Podstaw Informatyki PAN
	dr inż. Rafał	Angryk	Montana State University, USA
	dr hab. Zenon	Biniek	Wyższa Szkoła Technologii Informatycznych
	prof. dr hab. inż. Ryszard	Budziński	Zachodniopomorski Uniwersytet Technologiczny
	dr inż. Joanna	Chimiak-Opoka	University of Innsbruck, Austria
	prof. dr hab. inż. Ryszard	Choraś	Uniwersytet Technologiczno-Przyrodniczy
	dr inż. Grzegorz	Domek	Uniwersytet Kazimierza Wielkiego
	dr hab. inż. Piotr	Gajewski	Wojskowa Akademia Techniczna
	dr Marek	Holyński	Prezes Polskiego Towarzystwa Informatycznego
prof. dr hab. inż. czł. koresp. PAN	Janusz	Kacprzyk	Instytut Badań Systemowych PAN
	dr hab.inż. Andrzej	Kobyliński	Szkoła Główna Handlowa
	prof. dr.h.c.mult. Peter	Kopacek	Vienna University of Technology, Austria
prof. dr hab. inż. czł. koresp. PAN	Józef	Korbicz	Uniwersytet Zielonogórski
	prof. dr hab. Jacek	Koronacki	Instytut Podstaw Informatyki PAN
	prof. dr hab. Witold	Kosiński	Uniwersytet Kazimierza Wielkiego, PJWSTK
prof. dr hab. inż. Marek	Kurzyński		Politechnika Wrocławska
	dr hab.inż. Halina	Kwaśnicka	Politechnika Wrocławska
	prof. dr Mirosław	Majewski	New York Institute of Technology, United Arab Emirates
	dr hab. Andrzej	Marciniak	Politechnika Poznańska
	dr Marcin	Paprzycki	Instytut Badań Systemowych PAN
prof. dr hab. inż. Witold		Pedrycz	University of Alberta, Canada
prof. dr hab. inż. Andrzej		Piegat	Zachodniopomorski Uniwersytet Technologiczny
	dr hab.inż. Andrzej	Polański	Politechnika Śląska
prof. dr hab. inż. Orest		Popov	Zachodniopomorski Uniwersytet Technologiczny
	dr inż. Izabela	Rojek	Uniwersytet Kazimierza Wielkiego
prof. dr hab. inż. Danuta		Rutkowska	Politechnika Częstochowska
prof. dr hab. inż. Leszek		Rutkowski	Politechnika Częstochowska
prof. dr hab. inż. czł. koresp. PAN	Roman	Słowiński	Instytut Badań Systemowych PAN, Politechnika Poznańska
	dr hab. inż. Włodzimierz	Sosnowski	Uniwersytet Kazimierza Wielkiego, IPPT PAN
	prof. dr hab. inż. Andrzej	Stateczny	Akademia Morska w Szczecinie
	dr hab. Janusz	Szczepański	Uniwersytet Kazimierza Wielkiego, IPPT PAN
prof. dr hab. inż. czł. koresp. PAN	Ryszard	Tadeusiewicz	Akademia Górniczo-Hutnicza
prof. zw. dr hab. inż. czł. rzec. PAN	Jan	Węglarz	Instytut Chemii Bioorganicznej PAN, Politechnika Poznańska
	prof. dr hab. inż. Sławomir	Wierchoń	Instytut Podstaw Informatyki PAN
	dr hab. inż. Antoni	Wiliński	Zachodniopomorski Uniwersytet Technologiczny
	dr hab. inż. Andrzej	Wiśniewski	Akademia Podlaska
	dr hab. Ryszard	Wojtyna	Uniwersytet Technologiczno-Przyrodniczy
	dr hab. Felicja	Wysocka-Schillak	Uniwersytet Technologiczno-Przyrodniczy
	dr hab. Sławomir	Zadrożny	Instytut Badań Systemowych PAN
	dr hab. Zenon	Zwierzewicz	Akademia Morska w Szczecinie

**Studies and Materials  
in  
Applied Computer  
Science**

# Studies and Materials in Applied Computer Science

Journal of young researchers, PhD students and students

Endorsed by Polish Information Processing Society

## Chairman of Editorial Board

Janusz Kacprzyk Systems Research Institute, PAS

## Editors-in-Chief

Jacek Czerniak Kazimierz Wielki University in Bydgoszcz  
Marek Macko Kazimierz Wielki University in Bydgoszcz

## Secretary

Iwona Filipowicz Kazimierz Wielki University in Bydgoszcz

## Editorial Office

Mariusz Dramski Kazimierz Wielki University in Bydgoszcz  
Włodzimierz Masierak Kazimierz Wielki University in Bydgoszcz  
Piotr Prokopowicz Kazimierz Wielki University in Bydgoszcz  
Grzegorz Zych Kazimierz Wielki University in Bydgoszcz

## Editorial Board

Stanisław	Ambroszkiewicz	The Institute of Computer Science, PAS
Rafał	Angryk	Montana State University, USA
Zenon	Biniek	University of Information Technology
Ryszard	Budziński	The University of Szczecin
Joanna	Chimiak-Opoka	University of Innsbruck, Austria
Ryszard	Choraś	University of Technology and Life Sciences in Bydgoszcz
Grzegorz	Domek	Kazimierz Wielki University in Bydgoszcz
Piotr	Gajewski	Military University of Technology
Marek	Hołyński	President of Polish Information Processing Society
Janusz	Kacprzyk	Systems Research Institute, PAS
Andrzej	Kobyliński	The Warsaw School of Economics
Peter	Kopacek	Vienna University of Technology, Austria
Józef	Korbicz	University of Zielona Góra
Jacek	Koronacki	The Institute of Computer Science, PAS
Witold	Kosiński	Kazimierz Wielki University in Bydgoszcz, PJIT
Marek	Kurzyński	Wrocław University of Technology
Halina	Kwaśnicka	Wrocław University of Technology
Mirosław	Majewski	New York Institute of Technology, United Arab Emirates
Andrzej	Marciniak	Poznań University of Technology
Marcin	Paprzycki	Systems Research Institute, PAS
Witold	Pedrycz	Systems Research Institute, PAS
Andrzej	Piegat	West Pomeranian University of Technology
Andrzej	Polański	The Silesian University of Technology
Orest	Popov	West Pomeranian University of Technology
Izabela	Rojek	Kazimierz Wielki University in Bydgoszcz
Danuta	Rutkowska	Czestochowa University of Technology
Leszek	Rutkowski	Czestochowa University of Technology
Roman	Słowiński	Systems Research Institute, PAS, Poznań University of Technology
Włodzimierz	Sosnowski	Kazimierz Wielki University in Bydgoszcz, IFTR PAS
Andrzej	Stateczny	Maritime University of Szczecin
Janusz	Szczepeński	Kazimierz Wielki University in Bydgoszcz, IFTR PAS
Ryszard	Tadeusiewicz	AGH University of Science and Technology
Jan	Węglarz	Institute of Bioorganic Chemistry, PAS, Poznań University of Technology
Sławomir	Wierzchoń	The Institute of Computer Science, PAS
Antoni	Wiliński	West Pomeranian University of Technology
Andrzej	Wiśniewski	University of Podlasie
Ryszard	Wojtyna	University of Technology and Life Sciences in Bydgoszcz
Felicja	Wysocka-Schillak	University of Technology and Life Sciences in Bydgoszcz
Sławomir	Zadrozny	Systems Research Institute, PAS
Zenon	Zwierzewicz	Maritime University of Szczecin

**Studia i Materiały  
Informatyki  
Stosowanej**

czasopismo młodych pracowników  
naukowych, doktorantów i  
studentów

Tom 1, Nr 1, 2009

Bydgoszcz 2009

**Studia i Materiały Informatyki Stosowanej**  
czasopismo młodych pracowników naukowych, doktorantów i  
studentów

© Copyright 2009 by Uniwersytet Kazimierza Wielkiego

**ISSN 1689-6300**  
**ISBN 978-83-925180-4-4**

**Projekt okładki: Łukasz Zawadzki**  
**DTP: Nikodem Wojciechowski, Tomasz Sadłowski**

**Wydawca:**

Wydział Matematyki, Fizyki i Techniki  
Uniwersytet Kazimierza Wielkiego  
ul. Chodkiewicza 30  
85-064 Bydgoszcz  
tel. (052) 34-19-331  
fax. (052) 34-01-978  
*e-mail: simis@ukw.edu.pl*

**Kontakt:**

dr inż. Jacek Czerniak  
dr inż. Marek Macko  
Uniwersytet Kazimierza Wielkiego  
ul. Chodkiewicza 30  
85-064 Bydgoszcz  
*e-mail: jczerniak@ukw.edu.pl*  
*mackomar@ukw.edu.pl*

**Druk (ze środków sponsora):**  
**Oficyna Wydawnicza MW**

**Nakład 510 egz.**

**Bydgoszcz 2009**

## SPIS TREŚCI

<b>Słowo wstępne</b> .....	11
<b>W cyklu "Spotkania z Mistrzami":</b>	
<b>O potrzebie kształcenia oraz badań naukowych w obszarze Informatyki Stosowanej</b>	
RYSZARD TADEUSIEWICZ .....	15
<b>Nadesłane artykuły:</b>	
<b>Obiektowa implementacja algorytmu klasteryzacji metodą k-średnich</b>	
MARIUSZ DRAMSKI.....	25
<b>Implementacja liczb OFN i L-R w kalkulatorze porównawczym</b>	
IRENA KOPYSZKA .....	29
<b>Modele rywalizacji i walki w równaniach Lanchestera</b>	
MARCIN BAJEK .....	39
<b>Oscylator RLC z losowymi parametrami rezystancji i indukcyjności</b>	
PIOTR FAJKOWSKI.....	43
<b>Równanie van der Pola</b>	
PAWEŁ JENDYKIEWICZ.....	49
<b>Obliczenia ludzkie jako sposób na rozszerzanie baz wiedzy na podstawie gry 'Zgadnij autora'</b>	
TOMASZ SMYKOWSKI .....	55
<b>Idea skierowanych liczb rozmytych - Obserwacja rozmyta</b>	
DOROTA WILCZYŃSKA-SZTYMA.....	59
<b>Systemy mrówkowe w zastosowaniu do rozwiązania problemu komiwojażera</b>	
MIROŚLAW MODRZEJEWSKI .....	65
<b>Zaproszenie do publikowania</b> .....	77



## CONTENTS

<b>Editorial</b> .....	11
<b>Series 'Meetings with Masters':</b>	
<b>About Needs of Scientific Research and Education in Applied Computer Science</b>	
RYSZARD TADEUSIEWICZ .....	15
<b>Submitted Articles:</b>	
<b>Object oriented implementation of k-means clustering algorithm</b>	
MARIUSZ DRAMSKI.....	25
<b>Implementing OFN and L–R numbers in the comparative calculator</b>	
IRENA KOPYSZKA .....	29
<b>Competition and combat models in Lanchester equations</b>	
MARCIN BAJEK .....	39
<b>RLC oscillator with random parameters of resistance and inductance</b>	
PIOTR FAJKOWSKI.....	43
<b>Van der Pol equation</b>	
PAWEŁ JENDYKIEWICZ.....	49
<b>Human-based computation as a way to broaden knowledge bases</b>	
TOMASZ SMYKOWSKI .....	55
<b>The Idea of ordered fuzzy numbers – fuzzy observation</b>	
DOROTA WILCZYŃSKA-SZTYMA.....	59
<b>Ant Systems applied to solve the Travelling Salesman Problem</b>	
MIROSLAW MODRZEJEWSKI .....	65
<b>Call for Papers</b> .....	77



## SŁOWO WSTĘPNE

*Szanowni Państwo, Drodzy Czytelnicy.*

*Mamy zaszczyt i przyjemność przekazać Państwu pierwszy tom nowego periodyku naukowego Studia i Materiały Informatyki Stosowanej, wydawanego przez Wydawnictwo Uniwersytetu Kazimierza Wielkiego w Bydgoszczy. Pojawienie się tego czasopisma jest odpowiedzią na wyrażane często zapotrzebowanie na forum, gdzie mogliby publikować swoje prace informatycy polscy, głównie w języku polskim. Mamy nadzieję, że nasza oferta spotka się z życzliwym przyjęciem środowiska informatyków polskich, mimo istnienia na naszym rynku wydawniczym kilku czasopism naukowych z dziedziny informatyki.*

*Takie zapewnienia wyrażali liczni informatycy z wielu uczelni, instytutów naukowych i branżowych oraz przedstawiciele praktyki gospodarczej.*

*Z inicjatywą uruchomienia czasopisma wystąpiła grupa młodych ale już znanych informatyków bydgoskich, wśród których szczególnie trzeba wspomnieć dr. inż. Jacka Czerniaka i dr. inż. Marka Macko, którzy skupili wokół siebie silne i zaangażowane kolegium redakcyjne.*

*Idea ta zyskała życzliwe poparcie J.M. Rektora Uniwersytetu Kazimierza Wielkiego, prof. dr hab. inż. Józefa Kubika, Prorektora UWK ds. Nauki i Współpracy z Zagranicą, prof. dr hab. Janusza Ostojki-Zagórskiego, Dziekana Wydziału Matematyki, Fizyki i Techniki, dr. hab. Piotra Malinowskiego, prof. nadzw. UKW, Dyrektora Instytutu Mechaniki Środowiska i Informatyki Stosowanej, dr. hab. inż. Mariusza Kaczmarka, prof. nadzw. UKW, Dyrektora Instytutu Techniki, prof. dr, hab. inż. Bogusława Czupryńskiego oraz poprzedniego dyrektora Instytutu, prof. dr. hab. inż. Arnolda Wilczyńskiego. Wielki wkład w uruchomienie czasopisma wniosła także Dyrektor Wydawnictwa UKW, mgr Grażyna Jarzyna, oraz pracownicy Instytutu Mechaniki Środowiska i Informatyki Stosowanej, a w szczególności dr inż. Izabela Rojek i dr inż. Grzegorz Zych. Specjalne podziękowania należą się dr. inż. Jackowi Czerniakowi i dr. inż. Markowi Macko, Redaktorom Naczelnyim czasopisma.*

*Nasza idea zyskała poparcie czołowych informatyków polskich, którzy wyrazili chęć uczestniczenia w Radzie Naukowej.*

*Zamiarem Redakcji jest publikowanie oryginalnych artykułów w języku polskim i angielskim, przez uznanych specjalistów i młodych adeptów nauki. Ważną częścią naszego programu wydawniczego będą numery specjalne redagowane przez znanych autorów. Przewidujemy również zamieszczanie referatów przeglądowych przedstawiających stan wiedzy w wybranej dziedzinie, a także referatów specjalnych pisanych przez najwybitniejszych naukowców polskich i zagranicznych, prezentujących oryginalne, nierzadko kontrowersyjne, poglądy na wyzwania, jakie stawia przed szeroko rozumianą informatyką współczesny świat. Zamierzamy te specjalnie zamówione referaty prezentować w sekcji „Spotkania z Mistrzami”. Zależy nam bardzo, aby młodzi informatycy wkraczający na drogę naukową*

*Słowo wstępne*

*lub zawodową mogli poznać opinie znanych mistrzów dziedziny. Relacja „mistrz-uczeń”, obecna w nauce i kulturze od zarania dziejów, znajdzie więc w naszym czasopiśmie swoją kontynuację.*

*Mamy nadzieję, że nasza inicjatywa spotka się z życzliwym przyjęciem środowiska informatycznego w kraju i zagranicą i z góry dziękujemy Czytelnikom i Autorom za pomoc w budowaniu wizerunku i wysokiej pozycji naszego czasopisma.*

Prof. dr hab. inż. Janusz Kacprzyk  
Członek-korespondent PAN  
Fellow of IEEE  
Przewodniczący Rady Naukowej

*Serdeczne Podziękowania,*

*Chcielibyśmy podziękować wszystkim, dzięki którym czasopismo *Studia i Materiały Informatyki Stosowanej*, mogło zaistnieć. Przewodniczący Rady Naukowej wymienił już z nazwiska wiele zasłużonych osób. W pierwszej kolejności chcielibyśmy złożyć serdeczne podziękowania właśnie dla prof. dr. hab. inż. Janusza Kacprzyka, który mimo tak licznych obowiązków zechciał przyczynić się niezwykle aktywnie do powstawania nowego czasopisma naukowego. Jego niezwykły zapał i doświadczenie w pracach wydawniczych są dla nas niedoścignionym wzorem i ciągłą inspiracją. Następnie jeszcze raz dziękujemy Paniom i Panom Profesorom za włączenie się do Rady Naukowej SiMIS. W tym pierwszym numerze inaugurujemy też cykl „Spotkania z Mistrzami” i przy tej okazji chcieliśmy na ręce prof. dr. hab. inż. Ryszarda Tadeusiewicza, który jako pierwszy zdecydował się podzielić swoimi przemyśleniami, złożyć podziękowania wszystkim, którzy również zechcą ubogacić nas swoim słowem.*

*Jesteśmy również pełni wdzięczności dla Zarządu Głównego PTI z Prezesem dr. Markiem Hołyńskim na czele, za objęcie swoim patronatem SiMIS.*

*Dziękujemy gorąco wszystkim naszym młodym współpracownikom i studentom. Wasz zapał i determinacja w prowadzeniu badań naukowych staje się katalizatorem wielu działań, które podejmujemy, również po to aby ten młodzieńczy entuzjazm z jakim wdzieracie się w naukę nie został zbyt szybko zaduszony brakiem środków czy możliwości.*

*Mamy nadzieję że nasze wspólne plany wydawnicze uda się w tak zacnym zespole wypełnić i rozwinąć.*

Redaktorzy Naczelni SiMIS,  
Dr inż. Jacek Czerniak,  
Dr inż. Marek Macko



## **W CYKLU SPOTKANIA Z MISTRZAMI:**

### **O POTRZEBIE KSZTAŁCENIA ORAZ BADAŃ NAUKOWYCH W OBSZARZE INFORMATYKI STOSOWANEJ**

**Ryszard Tadeusiewicz**  
Akademia Górniczo-Hutnicza  
ul. Mickiewicza 30  
30-059 Kraków  
e-mail: [rtad@agh.edu.pl](mailto:rtad@agh.edu.pl)

**Streszczenie:**

Obecny system kształcenia w obszarze informatyki, mimo swoich bezspornych zalet, osiągnął w praktyce kres swojej wydolności, nie gwarantując jednocześnie możliwości wykształcenia takiej liczby informatyków, jaka będzie w najbliższym czasie potrzebna w związku z ogólnie znaną tendencją, określaną hasłowo jako zmierzanie do Społeczeństwa Informacyjnego. Istota przedstawionej w artykule koncepcji polega na tym, że zamiast naciągać do granic możliwości definicję studiów z zakresu "czystej informatyki" należy wyjść naprzeciw potrzebom gospodarki a także aspiracjom szturmującej bramy uniwersytetów młodzieży i utworzyć nowy kierunek studiów, któremu nadano nazwę Informatyka Stosowana. Zdefiniowano społeczną rolę absolwentów tego kierunku i pokazano, że jego powstanie jest cywilizacyjną koniecznością. Wskazano także na celowość prowadzenia badań naukowych w obszarze Informatyki Stosowanej. Przedyskutowano niektóre wiążące się z tym wątpliwości i konieczne do rozwiązania problemy. Jednak końcowy wniosek z tych dyskusji i rozważań jest jednoznaczny: Informatyka Stosowana jako kierunek kształcenia i jako obszar badań naukowych powinna być rozwijana.

**Słowa kluczowe:**

Informatyka stosowana, społeczeństwo informacyjne, edukacja informatyczna.

**Title: About Needs of Scientific Research and Education in Applied Computer Science**

**Abstract:**

Present system of education in computer science in Poland has many advantages, but not efficient enough, especially for new Information Society needs. We must educate more specialists in new (in Poland) branch of education, named Applied Computer Science. Idea of such education is subject of many controversies and discussions, therefore in the paper many arguments pro and contra are presented and discussed. Most important issue in this dispute is connected with social needs. Society need specialist who can serve as social interface between computer science professionals, discovering, developing and producing new hardware and software computer tools, and users, who adopt such products for many important practical proposes. The problem hot to educate Applied Computer Science specialists is not easy and many aspects of these education (and also connected scientific research) in Applied Computer Science are presented and discussed in the paper. But final conclusion is clear: Applied Computer Science should be developed both in the area of education and as a domain of scientific research.

**Keywords:**

Applied computer science, information society, education in computer science

## 1. WSTĘP

Debata na temat Informatyki Stosowanej toczy się od około dziesięciu lat. Jako osoba, która kiedyś wniosowała powstanie tego kierunku kształcenia, byłem i jestem bardzo żywo zaangażowany w tę dyskusję. Dlatego będąc zaproszonym do przedstawienia mojego artykułu w tym czasopiśmie, które ma Informatykę Stosowaną w swym tytule, pozwolę sobie przypomnieć pewne fakty i związane z nimi argumenty. Przedstawione tu poglądy są na wskroś subiektywne, nie aspirujące do tego, żeby przedstawiać fakty z wszystkich możliwych stron, dlatego w tekście używam pierwszej osoby liczby pojedynczej (zamiast przyjmowanego powszechnie w pracach naukowych bezosobowego formułowania wypowiedzianych zdań), bo nie usiłuję udawać, że to, co tu piszę, jest naukową prawdą obiektywną. Przeciwnie, jest to moje osobiste zdanie, co podkreślam dodatkowo, cytując w załączonej literaturze wyłącznie moje własne wcześniejsze publikacje na ten temat. Czynię tak mając nadzieję, że także i inne zainteresowane Osoby, w tym zarówno zwolennicy jak i przeciwnicy Informatyki Stosowanej zechcą przedstawić swoje poglądy i opinie, a ktoś mający do tej sprawy mniej zaangażowany stosunek zbuduje na tej podstawie opracowanie przedstawiające z większego dystansu bardziej zrównoważony całościowy sąd.

Najpierw spróbujmy uporządkować nieco „pole bitwy”. Wspomniana w poprzednim akapicie dyskusja dotyczy głównie tożsamości (względnie odrębności) Informatyki Stosowanej traktowanej jako kierunek studiów. Wiąże się z tym także oczywiste pytanie, czy Informatykę Stosowaną można traktować jako odrębny przedmiot badań naukowych, ale tu zacierzenie dyskutantów jest mniejsze i dlatego do tego zagadnienia odniosę się nieco później.

Co zatem możemy powiedzieć na temat celowości włączenia informatyki stosowanej do wykazu kierunków studiów, w ramach których możliwe jest kształcenie akademickie zgodne z wymaganiami procesu bolońskiego? Osobiście byłem i jestem gorącym zwolennikiem takiego włączenia. Zarysuję to w ujęciu chronologicznym, żeby pokazać (zgodnie z prawdą) formowanie się tego poglądu jako proces, a nie jako jednorazową decyzję.

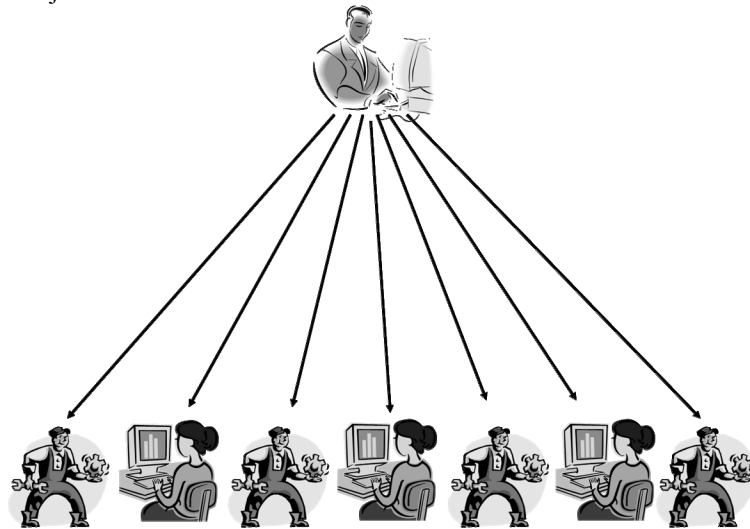
## 2. FORMOWANIE KONCEPCJI INFORMATYKI STOSOWANEJ JAKO KIERUNKU STUDIÓW

Pierwsze przesłanki wskazujące na celowość wprowadzenia do procesu kształcenia obszaru problemowego zastosowań informatyki (a nie tylko informatyki jako takiej) pojawiły się jeszcze w latach 80. ubiegłego stulecia [1]. Nie było wtedy jeszcze mowy o wprowadzeniu Informatyki Stosowanej jako nowego kierunku studiów, nie istniała także jeszcze (w języku polskim) odpowiednia nazwa, ale wielu nauczycieli akademickich dostrzegало już, że potrzeby gospodarcze i społeczne wymagają znacznie lepsze przygotowanie informatyczne specjalistów różnych branż (w tym zwłaszcza inżynierów), niż to dawały ówczesne studia [2]. Rodziło się przekonanie, że nie wystarczy doraźne przyuczanie na przykład mechaników czy elektryków do korzystania z technik komputerowych w procesie projektowania nowych urządzeń, a także ich kontroli, sterowania i eksploatacji, gdyż wolumen wiedzy z tym związanej bardzo szybko wzrastał, a sprawne korzystanie z odpowiednich informatycznych zasobów bywało ważniejsze w praktyce zawodowej od wiedzy ściśle profesjonalnej [3]. Z kolei liczne przykłady dowodziły także, iż brak odpowiedniej wiedzy informatycznej u nie-informatyków stawał się w wielu obszarach zastosowań komputerów silnym czynnikiem ograniczającym rozwój [4]. Dlatego pytanie postawione w 1996 roku: *Uczyć informatyki – ale jakiej?* [5] stawało się coraz bardziej pilne i ważne.

Pytanie to nabrało dodatkowego drammatyzmu w związku z faktem, że w tym samym okresie coraz powszechniej zaczęto mówić o tworzeniu Społeczeństwa Informacyjnego. Najpierw była to trochę utopijna idea, która pojawiła się w Japonii, USA oraz (w nieco innej postaci) we Francji, jednak pod koniec lat 90. stało się oczywiste, że Społeczeństwo Informacyjne jest bytem jak najbardziej realnym i że będzie musiało się uformować także w Polsce, gdyż był to jeden z oficjalnie zadekretowanych strategicznych celów Unii Europejskiej. Mając tak zdefiniowany cel rozwoju gospodarczego, społecznego i politycznego trzeba było zastanowić się, jak ten cel osiągnąć. Obok rozważań czysto teoretycznych rozpoczęto także badania modelowe [6, 7] z których wynikła między innymi taka obserwacja:

W Społeczeństwie Informacyjnym ukształtuje się nowy typ hierarchii, do którego kluczem będzie stosunek poszczególnych grup zawodowych (a nawet pojedynczych ludzi) do nowych technik informacyjnych (oznaczanych niekiedy skrótem ICT). Nie ulega wątpliwości, że informatycy tworzący i udostępniający społeczeństwu kolejne generacje narzędzi informatycznych będą tu rodzajem elity, określanej czasem w opracowaniach teoretycznych jako digitariat. Nieliczni, ale bardzo dobrze opłacani i mający ogromny wpływ na rozwój gospodarczy i społeczny członkowie digitariatu stanowiąc będą wierzchołek piramidy społecznego awansu. To będą (i do pewnego stopnia już teraz są) główni beneficjenci zachodzących przemian.

Nie ulega jednak wątpliwości, że w Społeczeństwie Informacyjnym będą także przegrani. Będą to wszyscy ludzie poprzestający na samym tylko „konsumowaniu” kolejnych osiągnięć informatyki bez żadnego własnego twórczego wkładu w tej dziedzinie. To oni formować będą podstawę tej piramidy, której wierzchołek tworzą członkowie digitariatu. Podstawę oczywiście bardzo rozległą i zróżnicowaną, dzielącą się zresztą dodatkowo na tak zwany kogitariat (grupę ludzi, którzy potrafią korzystać z udogodnień niesionych przez rewolucję informacyjną, ale sami nie uczestniczą w ich tworzeniu), oraz osób całkowicie nie korzystających z ICT, stanowiących proletariat<sup>1</sup> ery cyfrowej. Taki dwuwarstwowy model Społeczeństwa Informacyjnego wynika z wielu dyskusji teoretycznych toczonych na ten temat i jest przedstawiony poniżej na rysunku 1, pochodzącym z mojej prezentacji dotyczącej Informatyki Stosowanej, przedstawianej i dyskutowanej na 3. Kongresie Informatyki Polskiej w dniach 2-4 czerwca 2003 roku w Poznaniu.



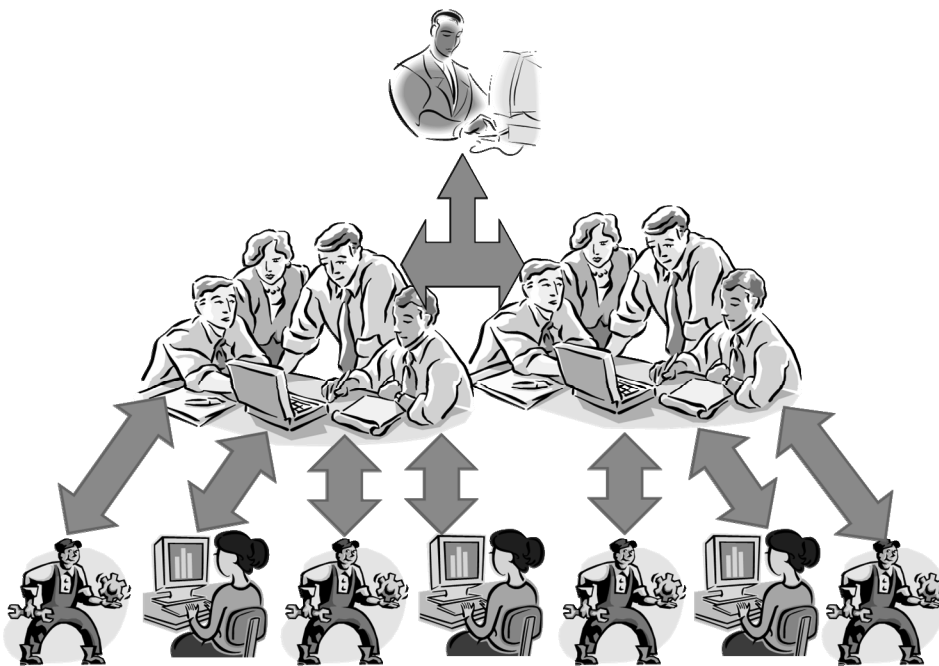
**Rys. 1.** Model Społeczeństwa Informacyjnego obejmujący wyłącznie twórców (digitariat) i prostych użytkowników informatyki (kogitariat i proletariat)

Nawet bardzo pobieżny rzut oka na schemat przedstawiony na tym rysunku budzi pewne zaniepokojenie. Pustka pomiędzy nieliczną elitą digitariatu, a ogromną rzeszą ludzi zepchniętych do roli kogitariatu i proletariatu wydaje się czymś nienaturalnym.

Zdefiniowanie i wprowadzenie profesji dla osób uprawiających informatykę stosowaną wypełnia tę lukę (rysunek 2).

---

<sup>1</sup> Nie bójmy się tego słowa: proletariat. Ma ono specjalny wydźwięk dla wszystkich, którzy przez wiele lat żyli w systemie komunistycznym, bo tam słowo to było ustawicznie używane (i nadużywane) w celach propagandowych. Jednak trzeba podkreślić, że proletariat ery Społeczeństwa Informacyjnego będzie bardzo odmienny od tego proletariatu wielkoprzemysłowego, którego niedole zainspirowały lewicową myśl społeczną i doprowadziły (ze szlachetnych pobudek!) do stworzenia socjalizmu jako teoretycznej ideologii sprawiedliwości społecznej, a potem do próby „siłowego” budowania realnego socjalizmu, który jednak w praktyce – jak wiadomo – skończył się gospodarczą i polityczną klęską.



Rys. 2. Model Społeczeństwa Informacyjnego uwzględniający dodatkową profesję: specjalistów Informatyki Stosowanej

Wprowadzenie zbiorowości specjalistów Informatyki Stosowanej tworzy układ strukturalnie i funkcjonalnie lepiej zrównoważony, niż wcześniej proponowany model z podziałem wyłącznie na digitalariat i kogitariat wraz z proletariatem. Co więcej, więzi pomiędzy poszczególnymi warstwami tego nowego modelu są znacznie silniejsze (co obrazują grubsze strzałki) oraz dwukierunkowe, co symbolizuje lepszy dwukierunkowy obieg informacji społecznej. Do zagadnienia tego powrócimy w dalszej części tego artykułu, nie kontynuując go w tym miejscu, bowiem ten rozdział ma na celu pokazanie (w ujęciu przekrojowym) jak doszło do formalnego zaistnienia Informatyki Stosowanej jako kierunku studiów.

Przywołując kolejne fakty związane z głównym tematem tego rozdziału warto stwierdzić, że nowy impuls w tej sprawie pojawił się na przełomie XX i XXI wieku, a dokładnie w 1999 roku. Wtedy właśnie jako powtórnie wybrany Rektor AGH, uczelni mającej renomę szkoły dobrze przygotowującej absolwentów do zawodu informatyka, zaobserwowałem zjawisko społeczne, koło którego (jak sądziłem) nie powinno się przechodzić obojętnie.

Chodziło mi o to, że wśród kandydatów na studia zapanowała wtedy trwająca do dziś „moda” na studiowanie informatyki, czego skutkiem była – między innymi – mordercza konkurencja przy rekrutacji. Przykładowo na moją uczelnię zgłaszało się rokrocznie ponad 1600 kandydatów chcących studiować informatykę, podczas gdy pojemności laboratoriów i inne uwarunkowania organizacyjne pozwalały na przyjęcie zaledwie 120 studentów na ten poszukiwany kierunek studiów. Oznaczało to, że rokrocznie ponad półtora tysiąca młodych ludzi, silnie motywowanych do tego, by w przyszłości wykonywać zawód związany z informatyką, było brutalnie pozbawianych tej szansy zaraz na początku ich samodzielnej aktywności zaraz po maturze. Dodam, że mówię tu o kandydatach doskonale przygotowanych do podjęcia upragnionych studiów, bowiem przeglądając statystyki stwierdzałem, że ci którzy „odpadali” w konkurencji na informatykę, mieli nieporównanie lepsze świadectwa maturalne i wyższe wyniki konkursowego egzaminu wstępnego, niż ci, których przyjmowaliśmy z otwartymi rękoma na niemal wszystkie pozostałe kierunki studiów.

Uważałem (i uważam nadal), że konieczne było wykonanie jakiegoś konkretnego kroku, wychodzącego naprzeciwko aspiracjom młodzieży pragnącej uzyskać wykształcenie informatyczne - w sytuacji, gdy nie było fizycznych możliwości ich kształcenia na „pękającym w szwach” kierunku „czystej informatyki”.

Dodatkowy asumpt do wykonania tego kroku opierał się na obserwacji (której także sprzyjała sprawowana funkcja rektora), że na wielu wydziałach proces dydaktyczny bardzo silnie związany był z wykorzystaniem w studenckich laboratoriach i przy pracach projektowych nowoczesnych technik komputerowych (na przykład komputerowo wspomagane projektowanie części maszyn albo komputerowe przetwarzanie obrazów preparatów metalurgicznych), zaś naukowcy zatrudnieni na tych wydziałach prowadzili badania w których komponenta informatyczna była często bardziej istotna, niż komponenta związana z określonym obszarem zastosowań (na przykład wiele prac w obszarze geoinformatyki).

### 3. PRÓBA UTWORZENIA NOWEGO KIERUNKU STUDIÓW – NIEPOWODZENIE GLOBALNE I SUKCES LOKALNY

W środowisku osób zajmujących się polskim systemem szkolnictwa wyższego zajmowałem wtedy dosyć prominentną pozycję, gdyż zostałem wybrany do sprawowania funkcji przewodniczącego KRPUT (Konferencji Rektorów Polskich Uczelni Technicznych) byłem wiceprzewodniczącym Konferencji Rektorów Krakowskich Szkół Wyższych oraz członkiem prezydium KRASP. Wykorzystując moje przemyślenia i opisane wyżej spostrzeżenia przygotowałem kompletny wniosek o utworzenie nowego kierunku studiów pod nazwą Informatyka Stosowana wraz z uzasadnieniem, zarysem minimów programowych, opisem sylwetki absolwenta itd. Dokument taki trzeba było skierować do Ministerstwa oraz do Rady Głównej (Państwowa Komisja Akredytacyjna jeszcze wtedy nie istniała, gdyż wniosek o jej powołanie sformułował KRASP nieco później). Wszystko to przygotowałem według najlepszych wzorów zagranicznych (gdzie kierunek kształcenia *Applied Computer Science* był od dawna obecny), a także skonsultowałem z kierownikiem jedynej w Polsce (w tamtych czasach) Katedry Informatyki Stosowanej Politechniki Łódzkiej, prof. Dominikiem Sankowskim, który dodatkowo był ekspertem Ministerstwa w sprawach związanych z programami studiów „zwykłej” informatyki. Wszystko było więc gotowe, przemyślane, uzgodnione, skorygowane. Biorąc pod uwagę powyższe uwarunkowania sądziłem, że stosunkowo łatwo przekonam zbiorowość polskich rektorów do tego, żeby przygotować wspólne wystąpienie w sprawie potrzeby utworzenia nowego kierunku studiów pod nazwą Informatyka Stosowana.

Mylilem się. Wprowadziłem sprawę pod obrady KRPUT w marcu 2002 roku i przekonałem się, że tak naprawdę nikogo to nie interesuje! Rektorzy wypowiadali się w sposób wymijający, uchwały nie dało się podjąć (była zasada, że dla uchwalenia czegoś powinien być konsensus, a tu go wyraźnie nie było), zaś w czerwcu 2002 roku w ramach wyboru władz nowej kadencji pozostałem nadal rektorem AGH, ale przestałem być przewodniczącym Konferencji Rektorów i musiałem każdą ogólnopolską akcję uzgadniać z moim następcą na stanowisku szefa KRPUT, który z zasady był przeciwny wszystkiemu, co pochodziło ode mnie.

Nie widząc szans na uzyskanie dla Informatyki Stosowanej statusu kierunku studiów prowadzonego i uznawanego w całej Polsce postanowiłem ratować, to co jeszcze było możliwe do uratowania: mając dobrze przygotowane plany i programy studiów na proponowanym kierunku studiów Informatyka Stosowana, a także mając studium możliwości (kadrowych, sprzętowych i lokalowych) podjęcia takich studiów na AGH – rozpocząłem starania o przyznanie prawa prowadzenia tego kierunku wyłącznie na mojej Uczelni. Ministerstwo początkowo nie chciało się zgodzić, ale wtedy przypuściłem ofensywę prasową: moje kolejne listy do Pani Minister (Krystyny Łybackiej) zacząłem traktować jako listy otwarte, publikując je w prasie [8, 9]. To poskutkowało i w 2003 roku AGH jako jedyna wyższa uczelnia w Polsce uzyskała prawo do kształcenia na kierunku Informatyka Stosowana. Tu warto podkreślić, że wiele uczelni potem wprowadziło Informatykę Stosowaną jako specjalność albo jako kierunek dyplomowania na różnych kierunkach studiów (na przykład dawny Wydział Matematyczno – Fizyczno – Chemiczny UJ, sąsiadujący z AGH przez ulicę Reymonta, przemianował się w 2007 roku na Wydział Fizyki, Astronomii i Informatyki Stosowanej), ale do dnia pisania tego artykułu (maj 2009) żadna polska uczelnia poza AGH nie dostała prawa kształcenia na kierunku Informatyka Stosowana.

W mojej Uczelni kształcenie na kierunku Informatyka Stosowana podjęły cztery wydziały: Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki, Wydział Metalurgii (który wkrótce potem zmienił nazwę na Wydział Inżynierii Metali i Informatyki Przemysłowej), Wydział Technicznej Fizyki Jądrowej (który wkrótce potem zmienił nazwę na Wydział Fizyki i Informatyki Stosowanej) oraz Wydział Geologii, Geofizyki i Ochrony Środowiska. Na każdym z tych wydziałów studenci kierunku Informatyka Stosowana stanowią najzdolniejszą i najprężniej działającą (koła naukowe, konferencje, prowadzenie stron WWW, wydawanie własnych czasopism itp.) część społeczności studenckiej, a sukcesy zawodowe pierwszych absolwentów stanowią najsilniejszą zachętę dla kolejnych pokoleń kandydatów do studiowania na tym kierunku. Mówiąc prawdę możliwość prowadzenia studiów na kierunku Informatyka Stosowana stanowiła dla tych wydziałów prawdziwie zbawienny impuls rozwojowy, gdyż zamiast malejącej liczby kandydatów na studia (co było ich udziałem wcześniej i co mogło się niebezpiecznie nasilić w związku z niżem demograficznym) notują one obecnie rosnące zainteresowanie kandydatów i są w stanie z roku na rok podnosić poziom wymagań stawianych swoim studentom. Również kadra tych wydziałów znalazła nowy obszar, w którym zdobywa sukcesy naukowe, a także uzyskuje stopnie i tytuły naukowe. Coraz więcej uczelni, które w momencie walki o utworzenie kierunku Informatyka Stosowana nie chciały się angażować i nie poparły moich starań – przysłała do nas swoich przedstawicieli, żeby zobaczyć, jak my to robimy, a w konsekwencji żeby ubiegać się o prawo kształcenia na kierunku Informatyka Stosowana.

#### 4. WIĘCEJ PYTAŃ NIŻ ODPOWIEDZI

Przytoczony wyżej skrócony opis historii utworzenia kierunku studiów Informatyka Stosowana na AGH był wart przytoczenia z dwóch powodów. Po pierwsze historia ta pokazuje, jak trudno jest „przebić się” z nową ideą, nawet jeśli idea ta jest trafna – a to, że była ona trafna potwierdza dość powszechne teraz dążenie do wprowadzenia Informatyki Stosowanej jako kierunku studiów nauczanego także na innych uczelniach. Po drugie mając obecnie dużą szansę na rozszerzenie tej inicjatywy na całą Polskę warto uświadomić wszystkim zwolennikom kształcenia na kierunku Informatyki Stosowanej, na jakie „rafy” nieuchronnie natrafiają. Bowiem pogląd, że po dziesięciu z górą latach walki o obecność Informatyki Stosowanej na edukacyjnej mapie Polski rafy te i przeszkody zniknęły - byłby zdecydowanie zbyt optymistyczny. Ten kierunek studiów ciągle jeszcze wzbudza kontrowersje, przy czym głównymi oponentami są zwykle profesjonalni informatycy, zwalczający Informatykę Stosowaną z różnych powodów, wśród których na plan pierwszy wysuwa się pogląd, że wprowadzenie tego nowego kierunku kształcenia deprecjonuje w jakimś stopniu informatykę „czystą” wprowadzając pewien zamęt na rynku pracy.

Daleki jestem od tego, żeby stanowczo twierdzić, iż oponenti całkowicie nie mają racji. Rzeczywistość jest (jak zawsze) wielobarwna, a nie tylko czarna lub biała, a porzekadło, że „diabeł tkwi w szczegółach” także nie daje o sobie zapomnieć. Szczegółów jest zaś w rozważanym tu problemie dużo – i nie da się ukryć, że są skomplikowane. Dlatego podczas przygotowywania programów dla kierunku Informatyka Stosowana (a także podczas praktycznej realizacji procesu kształcenia na tym kierunku) pojawiło się bardzo wiele pytań.

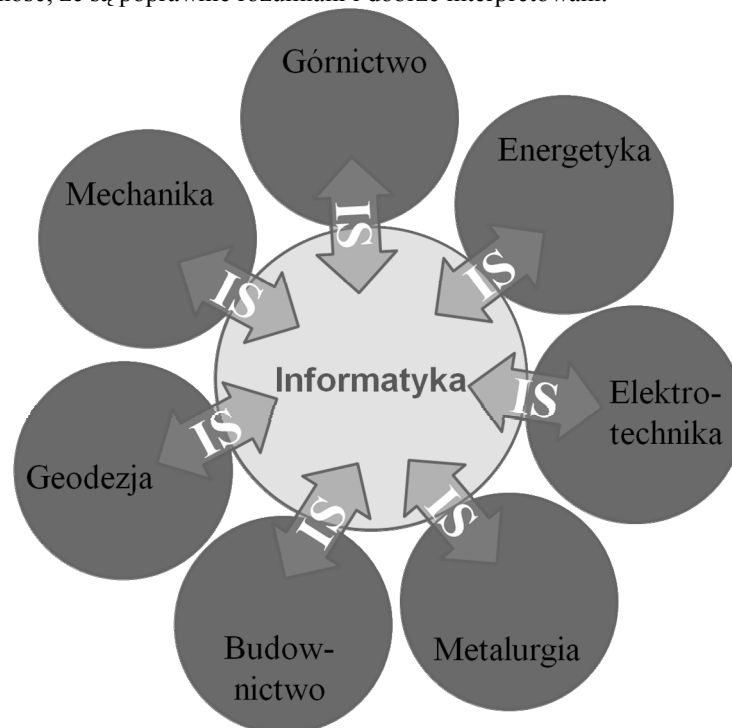
Pierwsze z nich dotyczyło zasobu wiedzy, jaką powinien się legitymować absolwent takiego kierunku. Nie ulega wątpliwości, że musi to być „specjalista hybrydowy”, mający sporą wiedzę z zakresu informatyki, ale także pewien zasób wiedzy z zakresu dziedziny, w której powinien w przyszłości tę wiedzę stosować. Pytanie, na które trzeba było odpowiedzieć dotyczyło tego, jakie wiadomości i umiejętności informatyczne powinien taki absolwent posiadać, żeby uczciwie można było o nim mówić, że jest specjalistą w zakresie informatyki (co prawda stosowanej, ale jednak informatyki). Dyskusje z informatykami nie przynosiły w tym zakresie oczekiwanych rozwiązań, ponieważ dominował w nich pogląd, że z absolutnie niczego, co składa się na minimum programowe studiów na kierunku informatyka (ta „normalna”) zrezygnować nie można – a to powodowało, że każda próba ułożenia sensownego planu i programu studiów dla Informatyki Stosowanej prowadziła do stwierdzenia, że brakuje co najmniej kilku semestrów w całym programie, kilku dodatkowych tygodni w każdym semestrze i kilku godzin powiększających dobę.

Ten sam problem, chociaż może nie z tak wielką intensywnością, pojawiał się w kontekście przedmiotów zawodowych, które powinny uzupełnić wykształcenie informatyczne o te aspekty praktyki – na przykład w zakresie inżynierii materiałowej albo geologii, żeby można było mówić, że absolwent będzie potrafił sensownie uczestniczyć w procesie stosowania informatyki w sobie właściwej dziedzinie.

Dla dodatkowego naświetlenia tego zagadnienia wróćmy do schematu pokazanego na rysunku 2, na którym w sposób ideowy zaznaczono umiejscowienie specjalistów Informatyki Stosowanej w strukturze Społeczeństwa Informacyjnego. Przy wprowadzaniu tego schematu zasygnalizowałem, że widoczny na nim układ strzałek łączących ze sobą symbolicznie zaznaczone grupy zawodowe jest celowo wyraźnie odmienny niż na rysunku 1 – ale tej różnicy wtedy nie komentowałem. Teraz warto wyjaśnić, jaka się w tym kryje myśl. Otóż rozwój Społeczeństwa Informacyjnego musi się opierać na komunikacji. W szczególności grupa informatyków (czystych) tworzących wierzchołek rozważanej piramidy, musi na bieżąco uzyskiwać i wykorzystywać wiadomości od użytkowników wytwarzanych przez nich narzędzi i zasobów informatycznych. Wiadomości te dotyczyć powinny ocen stworzonych narzędzi i wynikających z tych ocen dezyderatów związanych z ich doskonaleniem, a także konieczne jest pozyskiwanie wiadomości o tym, co jeszcze wymaga zainformatyzowania i jak można by było tę informatyzację przeprowadzić. Robotnik (czy nawet inżynier mechanik) sam nie ulepszy programu sterującego wielofunkcyjnym centrum obróbkowym, ale tylko on wie, w czym aktualne działanie tego centrum odbiega od idealnego modelu funkcjonowania w pełni zintegrowanego z innymi elementami technologicznego procesu. Z kolei informatyk potrafi stworzyć perfekcyjne oprogramowanie, które jednak może nie przystawać do potrzeb praktyki.

Możliwość bezpośredniego dogadania się informatyka z mechanikiem w modelu pokazanym na rysunku 1 jest problematyczna, bo nie tylko nie będą (wzajemnie) znali swoich problemów warsztatowych, ale w dodatku nie będą mieli wspólnego systemu odniesienia. Bariery, jakie tu powstają ujawniają się między innymi w postaci baraku obustronnie zrozumiałej terminologii, co jest wbrew pozorom poważniejszym problemem, niż się powszechnie sądzi. W modelu pokazanym na rysunku 2 te bariery i ograniczenia są usunięte dzięki temu, że specjaliści Informatyki Stosowanej znają problematykę (i język) informatyki, potrafią myśleć algorytmicznie, mogą skutecznie tworzyć poprawne specyfikacje określające wymagania dla systemów informatycznych – słowem potrafią dobrze postawić zadanie informatykom, którzy

chcą je rozwiązać. Dialog profesjonalnego informatyka, twórcy nowych systemów, z „informatykiem stosowanym” nie będzie przypominał dialogu głuchego ze ślepy – co niestety nader często ma miejsce w przypadku rozmowy informatyków ze specjalistami z określonych wąskich dziedzin zastosowań technik komputerowych. Aby jednak ten dialog mógł przynieść pożądane wyniki – celowe i konieczne jest by specjalista Informatyki Stosowanej potrafił z równą łatwością i profesjonalną znajomością rzeczy rozmawiać ze specjalistami określonej dziedziny zastosowań (na przykład mechanikami), bowiem tylko wtedy ci specjaliści dziedzinowi będą mogli mu przedstawiać swoje problemy, potrzeby i postulaty zachowując pewność, że są poprawnie rozumiani i dobrze interpretowani.

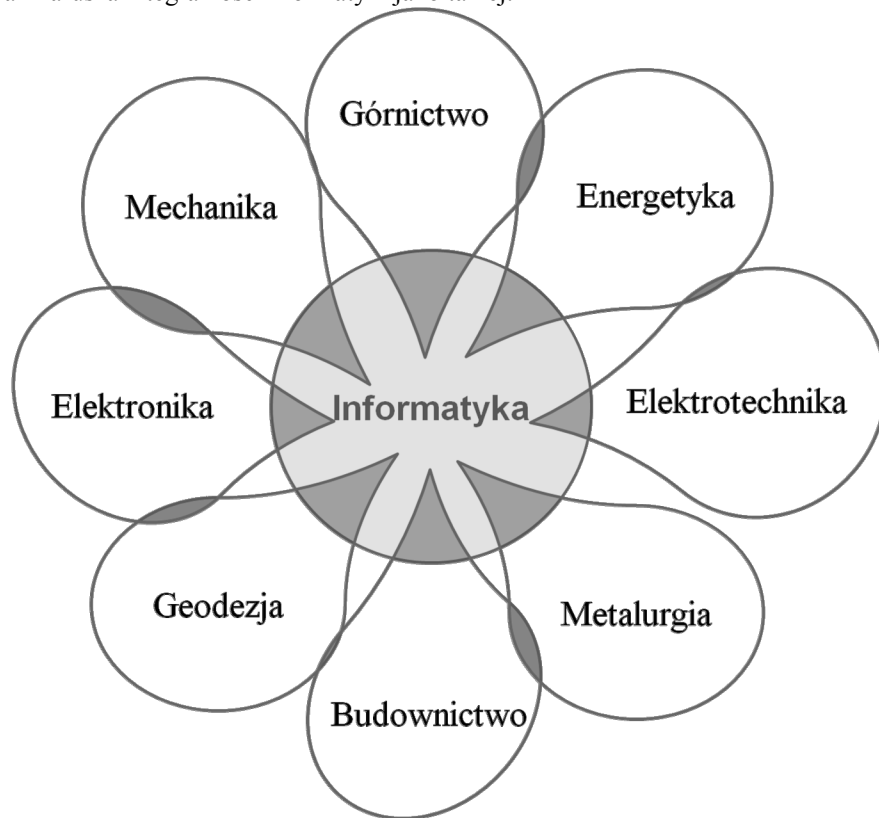


**Rys. 3.** Miejsce informatyki stosowanej (IS) jako dyscypliny sprzęgającej poszczególne dyscypliny szczegółowe z informatyką

Z przedstawionego rozumowania wynika, że rolą absolwentów Informatyki Stosowanej jest (między innymi) rola specyficznego społeczno-zawodowego „interfejsu” między profesjonalnymi informatykami z jednej i specjalistami dziedzinowymi z drugiej strony. To właśnie mają wyrażać grube dwukierunkowe strzałki na rysunku 2. Pozostaną one jednak tylko pobożnym życzeniem jeśli w strukturze kształcenia na kierunku Informatyka Stosowana nie zadba się o zrównoważone kompendium wiedzy zarówno informatycznej, jak i specjalistycznej. Ponieważ jednak nie da się zdobyć wiedzy specjalistycznej we wszystkich możliwych dziedzinach – jest oczywiste, że zawód specjalisty Informatyki Stosowanej będzie związany zawsze z jednym tylko „płatkiem kwiatu” pokazanego na rysunku 3. Dlatego celowe jest wprowadzenie kształcenia w tym potrzebnym (jak twierdzą) zawodzie na wielu różnych kierunkach studiów, w wyniku czego otrzymamy informatyków stosowanych o nachyleniu związanym z – przykładowo – mechaniką, elektrotechniką, geodezją, budownictwem, metalurgią, górnictwem itd. Wymieniłem tu tylko wachlarz specjalności związanych z naukami technicznymi, dosyć łatwo jednak do tego zestawu dodać na przykład ekonomię (gdzie odpowiednich specjalistów kształci się od lat na kierunku studiów Informatyka i Ekonometria, co także gniewa od dawna „czystych” informatyków), medycynę, pedagogikę (e-learning) i wiele innych. Informatyka jest już dziś (lub będzie w najbliższej przyszłości) obecna i służebna we wszystkich bez mała dziedzinach, więc potrzeby edukacyjne w tym obszarze są przeogromne!

Przy okazji rozważania postulowanej struktury i wzajemnych relacji kierunków kształcenia w systemie uwzględniającym obecność informatyki stosowanej (rysunek 3) warto zauważyć, że system tam przedstawiony jest zdrowszy i bardziej logiczny, niż system do jakiego obecnie kierunku kształcenia zwolna zmierzają, polegający na tym, że nieodzowność wiedzy informatycznej u wszystkich praktycznie specjalistów wszystkich dziedzin powoduje, że każda z tych dziedzin z osobna wykształca w swojej strukturze „wypustkę” wdzierającą się do obszaru informatyki i asymilującą

mniej lub bardziej skutecznie różne dobrodziejstwa techniki komputerowej w kontekście swoich partykularnych potrzeb. System ten umownie przedstawiono na rysunku 4 pokazując, że taki model rozwoju deformuje korzystające z informatyki dziedziny kształcenia i narusza integralność informatyki jako takiej.



**Rys. 4.** Model ewolucji różnych kierunków kształcenia przy braku informatyki stosowanej

Przywołane wyżej problemy nie są jedynymi kwestiami, jakie podnosi się w kontekście informatyki stosowanej. Przykładowo debatuje się także, czy na kierunku studiów Informatyki Stosowanej (aktualnie oficjalnie w skali ogólnopolskiej nie istniejącym!) powinno być możliwe kształcenie wyłącznie w ramach studiów pierwszego stopnia, czy dopuścić oba stopnie, czy może właśnie powinno to być wyłącznie kształcenie na studiach drugiego stopnia po ukończeniu studiów pierwszego stopnia w ramach tradycyjnego systemu kierunków studiów, co da solidne podstawy fachowe w zakresie jednej z dziedzin aplikacji technik komputerowych przez przyszłego specjalistę Informatyki Stosowanej. Niektórzy dyskutujący są skłonni także dołączać tu trzeci stopień (studia doktoranckie), co jednak związane jest bezpośrednio z problemem, czy w zakresie Informatyki Stosowanej można prowadzić na tyle wyraźnie zdefiniowane badania naukowe, żeby możliwe było nadawanie stopni i tytułów naukowych w tym obszarze, traktowanym tym razem jako odrębna dyscyplina naukowa? Zajmiemy się tym w kolejnym rozdziale.

## 5. BADANIA NAUKOWE W DYSCYPLINIE INFORMATYKI STOSOWANEJ

Zajmując się planami kształcenia na kierunku Informatyki Stosowanej bywałem często indagowany w sprawie badań naukowych, jakimi powinni się legitymować nauczyciele akademicy, którzy tworzyć będą kadre dla tego kierunku. Zgodnie z obowiązującym prawem, a także zgodnie z zasadami, które wprowadził do koncepcji nowoczesnego uniwersytetu Alexander Heinrich Friedrich von Humboldt – nauczyciel akademicki powinien prowadzić aktywne badania naukowe w dyscyplinie, której naucza studentów. Od sukcesów odnoszonych w tych badaniach zależą stopnie tytuły naukowe nauczającej kadry, a posiadanie odpowiedniej liczby uczonych mających stosowne stopnia i tytuły jest warunkiem koniecznym uzyskania uprawnień do kształcenia na określonym kierunku studiów. Co więcej, wymaga się (i

jest to słuszne!) by zapewniona była zgodność pomiędzy kierunkiem studiów, na którym prowadzone jest nauczanie, a dyscypliną naukową, w której „firmujący” kształcenie naukowcy uzyskali swoje stopnie i tytuły naukowe.

Dla większości tradycyjnych kierunków studiów sytuacja jest jasna i jednoznaczna. Istnieje na przykład kierunek kształcenia Mechanika i istnieje dyscyplina naukowa Mechanika, więc jest także wielu uczonych legitymujących się stopniami i tytułami naukowymi w tej właśnie dyscyplinie. Ustalenie, czy określony wydział określonej uczelni może kształcić magistrów inżynierów mechaników sprowadza się więc do elementarnego policzenia nazwisk na liście Rady Wydziału.

W przypadku kształcenia na kierunku Informatyka Stosowana sprawa nie jest taka prosta, bowiem pojawia się pytanie, jaka część spośród „firmujących” te studia naukowców powinno legitymować się dorobkiem naukowym w dyscyplinie odpowiadającej obszarowi zastosowań (a więc na przykład wspomnianej wyżej mechanice), a ilu powinno być informatyków? Z tymi ostatnimi bywa zresztą kłopot, bo często prowadzą oni badania naukowe interdyscyplinarne (twórczo stosując informatykę w dziedzinie, która jest wybranym przez nich obszarem zastosowań), jednak prace tego typu trudno jest obronić w dyscyplinie informatyka (no bo rzeczywiście informatyki one naukowo nie wzbogacają), dlatego specjaliści tacy mają zwykle stopnie i tytuły raczej związane z wybranym obszarem zastosowań (na przykład mechaniką), a nie z obszarem informatyki, co powoduje, że ich uwzględnienie przy staraniach o prawo do dyplomowania w zakresie Informatyki Stosowanej bywa kwestionowane. Problem jest zresztą szerszy. Coraz więcej badaczy gromadzi znaczący dorobek naukowy niebanalnie stosując komputery w coraz to nowych obszarach i kontekstach. Gdyby stosowali na przykład woltomierze, to by zapewne mogli się ubiegać o uznanie ich prac za wartościowe w środowisku metrologów. Ale oni stosują komputery – a środowisko informatyków jest wyjątkowo hermetyczne. Dlatego z każdym tego typu przypadkiem wiąże się bardziej lub mniej ostra dyskusja: przyznać za taką pracę doktorat w informatyce, czy nie przyznać? Co ona wnosi do informatyki? Czy te badania wzbogacają zasób wiedzy informatycznej? Znam te polemiki już na pamięć. Dotyczy to nie tylko nadawania stopni naukowych, ale także na przykład przyznawania grantów. Przez dwie kadencje byłem członkiem KBN, potem przez wiele lat przewodziłem Sekcji Informatyki w Ministerstwie Nauki, a obecnie jestem członkiem Rady Nauki. Widziałem mnóstwo niezwykle ciekawych projektów prac badawczych, które jednak nie mogły uzyskać finansowania, ponieważ były ulokowane po części w informatyce, a po części w jakiejś innej dyscyplinie, dla której te informatyczne narzędzia pracowały. Bywało tak, że wszyscy oceniający zgadzali się, że praca jest wartościowa i powinna być finansowana, ale każdy chciał, żeby finansowała to ta druga strona (tzn. informatycy wpychali to – na przykład – mechanikom, a tamci odsyłali to do informatyków). Dlatego twierdzę, że powinna zostać zdefiniowana Informatyka Stosowana jako dyscyplina badań naukowych. Wtedy będzie możliwość właściwego rozwiązywania takich dylematów.

Z przeprowadzonej dyskusji wynika, że Informatyka Stosowana jest nam potrzebna zarówno jako kierunek kształcenia, jak i jako dyscyplina naukowa. Według mojej oceny nie ma już miejsca na pytanie czy takie prawo obywatelstwa Informatyce Stosowanej nadać. Jedyne aktualne pytanie brzmi: kiedy? Moja odpowiedź jest jednoznaczna: Jak najszybciej!

## **6. LITERATURA**

- Tadeusiewicz R., Nauczanie podstaw informatyki na studiach wyższych. *Życie Szkoły Wyższej*, nr 7-8, 1980, pp. 123-133
- Tadeusiewicz R., Informatyka, czyli jak tego robić nie należy. Artykuł w pracy zbiorowej: „Wobec największych zagrożeń”, część III "Dylematy nauki i nauczania", pp. 144 - 157, TWWP, Kraków 1988, pp. 245-266
- Tadeusiewicz R.: Nauczanie informatyki. *Informatyka*, nr 2, 1987, pp. 22 - 25
- Tadeusiewicz R., Rola edukacji informatycznej w rozwoju zastosowań komputerów. Społeczne uwarunkowania zastosowań informatyki w zarządzaniu, TNOiK, AE, Kraków, 1986, pp. 1 - 12
- Tadeusiewicz R., Uczyć informatyki – ale jakiej? *Kultura i Edukacja* nr 1, 1996, pp. 147-154
- Tadeusiewicz R., Model społeczeństwa informacyjnego, *Forum Akademickie*, nr 12, 1998, pp. 28-30
- Tadeusiewicz R., Development and studying of a small model of information society at the University of Mining and Metallurgy, In *Proceedings of Second International Conference on Research for Information Society*, National Institute of Telecommunications, 1999, Vol. A, pp. 64-68 (abstract) and Vol. B, part 12, pp. 1/12 – 12/12 (full text with bibliography)
- Tadeusiewicz R., List w sprawie informatyki stosowanej, *Biuletyn Informacyjny Pracowników AGH*, nr 114, luty 2003, ss. 4-5
- Tadeusiewicz R.: W sprawie informatyki. *Dziennik Polski* nr 58 (17 851), 10.03.2003, str. 24

*W cyklu „Spotkania z Mistrzami”: Ryszard Tadeusiewicz, O potrzebie kształcenia oraz badań naukowych...*

## NADEŚLANE ARTYKUŁY

# OBIEKTOWA IMPLEMENTACJA ALGORYTMU KLASTERYZACJI METODĄ K-ŚREDNICH

**Mariusz Dramski**

Uniwersytet Kazimierza Wielkiego  
Instytut Techniki  
ul. Chodkiewicza 30/p.215, 85-064 Bydgoszcz  
e-mail: mdramski@ukw.edu.pl

**Streszczenie:** *Algorytm klasteryzacji metodą k-średnich to jeden z najpopularniejszych sposobów służących do klasyfikacji danych przy użyciu metod sztucznej inteligencji. Otrzymane klastry mogą dalej posłużyć do budowy np. modeli neuronowych z wykorzystaniem dzwonów Gaussa (sieci RBF) czy rozmytych modeli Takagi-Sugeno. Niniejszy artykuł przedstawia implementację tego algorytmu w języku C++. Można tu znaleźć opis klasy, która może później posłużyć jako biblioteka do dowolnego programu napisanego w tym języku.*

**Słowa kluczowe:** *Klasteryzacja, metoda k-średnich, klasyfikacja danych, C++*

## Object oriented implementation of k-means clustering algorithm

**Abstrakt:** *K-means clustering algorithm is one of the most popular ways for data classification using artificial intelligence methods. Obtained clusters can be further used e.g. to build RBF networks or Takagi-Sugeno fuzzy models. This paper contains the implementation of this algorithm in C++ programming language. You can find there the description of the class, which can serve as a library in different programs written in C++.*

**Keywords:** *Clustering, k-means method, data classification, C++*

### 1. WSTĘP

Zadanie klasteryzacji polega na podzieleniu zbioru danych pomiarowych na pewne podzbiory, spełniające określone warunki. Takie podzbiory nazywamy klastrami. Najczęstszym stosowanym tutaj kryterium jest odległość w metryce euklidesowej. Otrzymane klastry mogą potem posłużyć do budowy lokalnych modeli systemów. Modele takie budujemy, kiedy nie interesuje nas przestrzeń całego systemu, bądź wówczas kiedy takie podejście jest wydajniejsze. Można również zbudować kilka modeli lokalnych, a za ich pomocą stworzyć jeden model globalny.

### 2. METODA K-ŚREDNICH

Jednym z najczęściej stosowanych metod klasteryzacji jest metoda k-średnich. Charakteryzuje się ona dość prostym aparatem matematycznym. Składa się ona z następujących kroków:

I. Przyjęcie klastrów startowych, ich środki wyznaczone są z reguły empirycznie, również na podstawie oceny wizualnej.

II. Przyporządkowanie próbek pomiarowych do poszczególnych klastrów na podstawie odległości euklidesowych  $d_{ij}$  poszczególnych próbek  $P_i$  od środków klastrów  $m_j$ .

$$d_{ij} = \|x_j - m_i\| = \sqrt{\sum_{l=1}^2 (x_{l,j} - m_{x_{l,j}})^2}$$

III. Określenie nowych środków klastrów, robi się to metodą skumulowaną na podstawie wzoru:

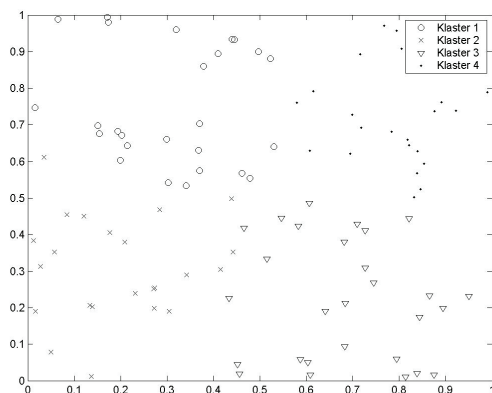
$$m_{x_{l,i}}(1) = \frac{1}{N_i} \sum_{j=1}^{N_i} x_{l,j}(0)$$

IV. Określenie wielkości przesunięcia klastrów  $\Delta m$

$$\Delta m = \|m_i(0) - m_i(1)\|$$

V. Przyporządkowanie próbek pomiarowych do nowych klastrów.

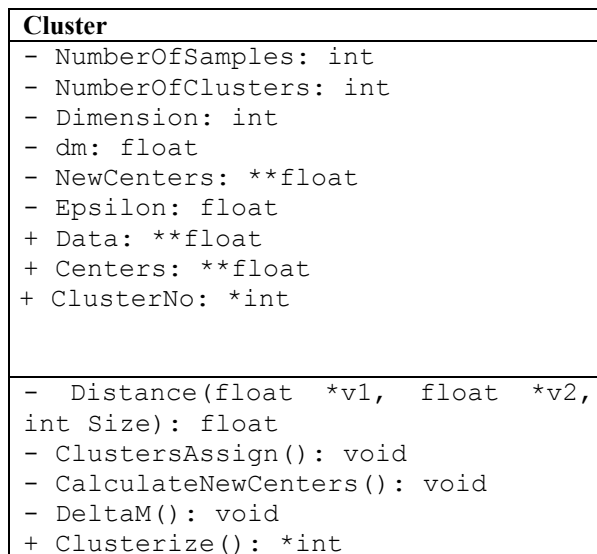
VI. Określenie nowych środków klastrów; powyższe wszystkie operacje powtarza się aż do momentu, w którym przesunięcie środków klastrów nie będzie przekraczało pewnej zadanej wartości progowej  $\epsilon$ .



Rysunek. 1 Przykładowa ilustracja algorytmu klasteryzacji metodą k-średnich, źródło: opracowanie własne.

### 3. IMPLEMENTACJA ALGORYTMU W JĘZYKU C++

W celu zaimplementowania algorytmu klasteryzacji metodą k-średnich, zaprojektowano wzorzec klasy, który można przedstawić za pomocą poniższego diagramu UML:



Jak łatwo zauważyć, atrybutami publicznymi klasy Cluster są jedynie dane pomiarowe, współrzędne środków klastrów oraz wektor zawierający numery klastrów dla każdej kolejnej próbki.

Jedyną publiczną funkcją jest funkcja Clusterize(), która uruchamia całą procedurę obliczeniową.

Korzystanie z klasy Cluster jest bardzo proste. Na początku należy wprowadzić macierz, zawierającą dane pomiarowe (atrybut Data). Następnie podaje się pierwotne środki klastrów (atrybut Centers). Ostatnim krokiem jest uruchomienie funkcji Clusterize().

Przykładowy kod źródłowy, wykorzystujący klasę Cluster, może zatem wyglądać następująco:

```
Cluster c(b1,b2,data,centers);
c.Clusterize();
```

Wektor b1 zawiera kolejno liczbę próbek, liczbę klastrów oraz liczbę wejść systemu. W wektorze b2 zawarte są informacje na temat  $\Delta m$  oraz  $\epsilon$ .

Wykonanie programu polegać będzie zatem na powołaniu do życia obiektu klasy Cluster, pamiętając o nadaniu mu stosownych wartości atrybutów i to już na poziomie konstruktora klasy. Nie istnieje potrzeba np. dalszej aktualizacji danych. Następną operacją jest uruchomienie funkcji Clusterize() na rzecz utworzonego obiektu. Funkcja ta korzysta z pozostałych funkcji klasy, które znajdują się w jej części prywatnej. Jej kod źródłowy wygląda następująco:

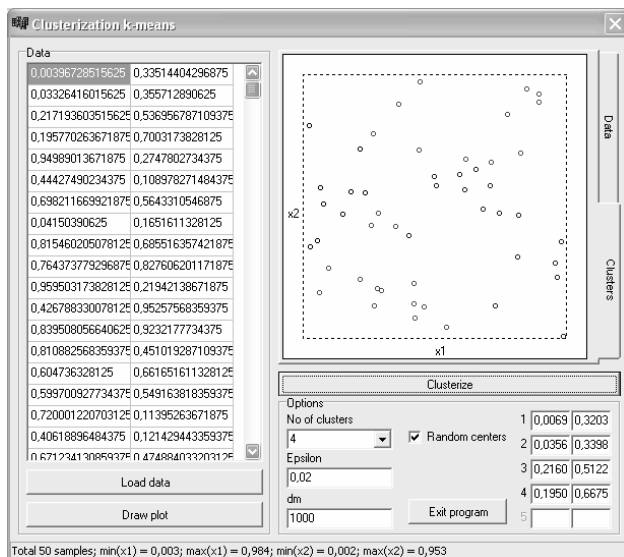
```
int* Cluster::Clusterize()
{
    ClustersAssign();
```

```

while (dm > Epsilon)
{
    CalculateNewCenters();
    for (int i=0; i<NumberOfClusters;
i++)
        for (int j=0; j<Dimension; j++)
            Centers[i][j] =
NewCenters[i][j];
            DeltaM();
            ClustersAssign();
        }
    return ClusterNo;
}

```

Funkcja Clusterize() zwraca jako rezultat wektor ClusterNo, który zawiera informacje na temat klastra odpowiadającego każdej próbce ze zbioru danych pomiarowych (kolejny numer próbki to po prostu indeks pokazujący jej pozycję w wektorze).



**Rysunek. 2** Interfejs przykładowej aplikacji wykorzystującej klasę Cluster.

Powyższy rysunek przedstawia interfejs przykładowej aplikacji wykorzystującej klasę Cluster. Jej celem jest zilustrowanie działania algorytmu klasteryzacji metodą k-średnich. Wynik klasteryzacji przedstawiony jest za pomocą kolorów. Każdemu klastrowi odpowiada inny kolor.

Taka aplikacja może służyć nie tylko do zademonstrowania działania algorytmu, ale może być także podstawą do tworzenia innych, bardziej skomplikowanych programów np. do modelowania systemów za pomocą sieci RBF.

#### 4. PODSUMOWANIE I WNIOSKI

W niniejszym artykule przedstawiono algorytm klasteryzacji metodą k-średnich, a także jego implementację w języku C++. Nie zamieszczono kodu źródłowego niektórych funkcji, ale są one stosunkowo proste do odtworzenia, dlatego nie istnieje po prostu taka potrzeba. Dla zainteresowanych autor udostępni pełny kod źródłowy drogą mailową.

Klasteryzacja metodą k-średnich to stosunkowo prosty algorytm służący do klasyfikacji danych. Jak widać jest on również łatwy do implementacji.

#### Literatura

1. Broomhead S., Lowe D., „Multivariable fractional interpolation and adaptive network”, *Complex Systems* 2, 321-323, 1988
2. Dramski M., „Opracowanie metody modelowania systemów nieliniowych z wykorzystaniem sąsiedztwa okrężnego”, rozprawa doktorska, Politechnika Szczecińska, 2005
3. Grębosz J., „Symfonia C++”, Oficyna Kalimach, Kraków 1999
4. Moody D., Darken J., “Fast learning in networks of locally-tuned processing units, *Neural Computation* 1, 281-294, 1988
5. Piegat A., “Modelowanie i sterowanie rozmyte”, Akademicka Oficyna Wydawnicza Exit, Warszawa 1999
6. Takagi T., Sugeno M., „Fuzzy identification of systems and its applications to modeling and control”, *IEEE Transactions on Systems, Man, and Cybernetics* 1985, vol. 15, No. 1, pp 116-132.



## IMPLEMENTACJA LICZB OFN I L-R W KALKULATORZE PORÓWNAWCZYM

Irena Kopyszka

Uniwersytet Kazimierza Wielkiego  
Instytut Techniki  
II rok MU Edukacja Techniczno-Informatyczna  
ul. Chodkiewicza 30, 85-064 Bydgoszcz  
e-mail: ikzlot@op.pl

**Streszczenie:** *Artykuł ten pokazuje znaczenie logiki rozmytej w życiu codziennym. Zawiera on podstawowe pojęcia z tej dziedziny. Wskazuje praktyczne wykorzystanie teorii zbiorów rozmytych. Porównuje jej arytmetykę z arytmetyką skierowanych liczb rozmytych. Opracowany program Skier\_LR w wyniku testowania pozwala na sformułowanie wniosku: skierowane liczby rozmyte pozwalają poszerzyć dziedzinę rozwiązań w porównaniu z klasycznym ujęciem liczb rozmytych.*

**Słowa kluczowe:** *Liczba rozmyta, skierowana liczba rozmyta, logika rozmyta*

### Implementing OFN and L–R numbers in the comparative calculator

**Abstract:** *This paper shows the importance of fuzzy logic in everyday live. It includes basic definitions of that field of science. It also indicates to practical applications of fuzzy sets theory. It includes comparison of the fuzzy sets arithmetic with arithmetic of ordered fuzzy numbers. Tests on Skier\_LR program permit to conclude that: with fuzzy numbers it is possible to expand the scope of solutions in comparison to fuzzy numbers in classic form.*

**Keywords:** *fuzzy number, ordered fuzzy number, fuzzy logic*

#### 1. WPROWADZENIE

CieŜko jest nam wyobrazić sobie życie bez takich urządzeń jak: pralka, lodówka, komputer czy też inne urządzenia codziennego użytku. Nie wszyscy jednak wiedzą, że podstawą ich działania jest logika rozmyta (*ang. fuzzy logic*) czyli operacje na nieprecyzyjnych danych (ciepło, zimno, szybko, „prawie 5” itp.).

Twórcą *Fuzzy logic* jest amerykański matematyk, profesor Uniwersytetu Columbia w Nowym Jorku i Berkeley w Kalifornii – Lotfii Askar Zadeha, który w roku 1965 w czasopiśmie „Information and Control” opublikował artykuł „Fuzzy sets”. Zdefiniował w nim pojęcie zbioru rozmytego, dzięki czemu nieprecyzyjne dane mogły zostać opisane wartościami z przedziału (0,1). Przypisana liczba oznacza ich stopień przynależności do tego zbioru. Warto przy tym wspomnieć, że L.Zadeh w swojej teorii wykorzystał

artykuł dotyczący logiki trójwartościowej, opublikowany 45 lat wcześniej przez Polaka – Jana Łukaszewicza, Dlatego też wielu naukowców na świecie uważa właśnie tego Polaka za „ojca” logiki rozmytej.

Dziesięć lat później teoria ta znalazła zastosowanie po raz pierwszy. W 1975 roku E.Mamdani opisał i zbudował pierwszy układ sterowania.

Największe zainteresowanie zyskała logika rozmyta w Japonii. Stała się ona podstawą projektu układu sterowania metra Sendai, tutaj też znalazła zastosowanie w urządzeniach domowych takich jak: pralki, lodówki, kamery video oraz w wielu dziedzinach gospodarki. Gwałtowny rozwój techniki w Japonii, opartej na logice rozmytej, spowodował zainteresowanie się tym tematem naukowców z innych krajów. Wydano wiele publikacji dotyczących tego zagadnienia i opracowano wiele programów wykorzystujących modelowanie i sterowanie rozmyte.

Duży wkład w światowy rozwój teorii zbiorów rozmytych wnieśli polscy naukowcy m.in. prof. E. Czogała, prof. J. Kacprzyk czy też prof. W. Pedrycz. Witold Kosiński, profesor Uniwersytetu Kazimierza Wielkiego w Bydgoszczy oraz w Polsko-Japońskiej Wyższej Szkole Technik Komputerowych w Warszawie, rozwinął tę teorię o **skierowane liczby rozmyte** (OFN). Porównując arytmetykę zbiorów rozmytych i skierowanych liczb rozmytych można jednoznacznie stwierdzić, że skierowane liczby rozmyte pozwalają poszerzyć dziedzinę rozwiązań w porównaniu z klasycznym ujęciem liczb rozmytych. Działania wykonywane na nich są mniej skomplikowane a uzyskane wyniki w większości przypadków dokładniejsze.

### 5. PODSTAWOWE POJĘCIA TEORII SKIEROWANYCH LICZB ROZMYTYCH

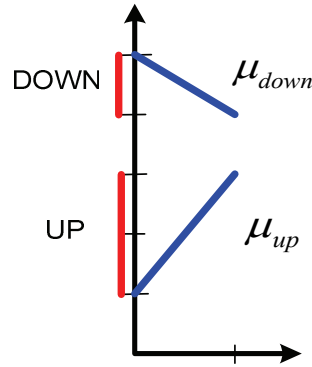
Każde działanie na liczbach rozmytych niezależnie od tego czy jest to dodawanie, odejmowanie, dzielenie czy mnożenie powoduje zwiększenie nośnika. W skutek wykonania kilku działań na danych liczbach L-R, można otrzymać liczby zbyt szerokie przez co mogą się stać mniej użyteczne. Także rozwiązywanie równań za pomocą klasycznych działań na liczbach rozmytych zazwyczaj jest nie możliwe. Równanie typu  $A+X = C$  można rozwiązać za pomocą klasycznych działań na liczbach rozmytych tylko w przypadku, gdy  $A$  jest liczbą rzeczywistą.

Pierwsze próby przededefiniowania działań na liczbach rozmytych podjęte zostały w 1993 roku przez profesora Witolda Kosińskiego i jego doktoranta – P. Słyszka. Dalsze publikacje prof. W. Kosińskiego przy współpracy z P. Prokopowiczem i D. Ślęzakim doprowadziły do wprowadzenia modelu **skierowanych liczb rozmytych** (ang. **ordered fuzzy numbers – OFN**).

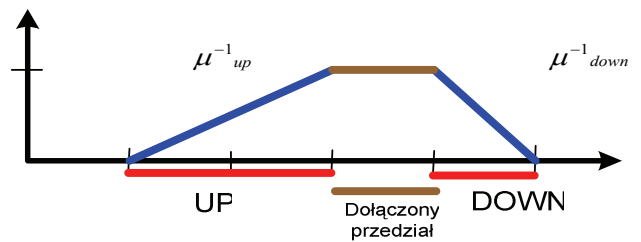
**Skierowaną liczbą rozmytą**  $A$  nazywamy uporządkowaną parę funkcji

$$A = (x_{up}, x_{down}),$$

gdzie  $x_{up}, x_{down} : [0,1] \rightarrow R$  są funkcjami ciągłymi. Odpowiednie części funkcji nazywamy: częścią **up** i **down**.



Rysunek 1. Skierowana liczba rozmyta



Rysunek 2. OFN przedstawiona w sposób nawiązujący do liczb rozmytych

Z ciągłości obydwu części wynika, że ich obrazy są ograniczone przedziałami. Odpowiednio otrzymują nazwy:  $UP$  i  $DOWN$ .

Dla oznaczenia granic (będących liczbami rzeczywistymi) tych przedziałów przyjęto następujące oznaczenia:  $UP = (l_A, 1_A^-)$  oraz  $DOWN = (1_A^+, p_A)$ . Jeżeli obydwie funkcje będące częściami liczby rozmytej, są ściśle monotoniczne to istnieją dla nich funkcje odwrotne  $x_{up}^{-1}$  i  $x_{down}^{-1}$  określone na odpowiednich przedziałach  $UP$  i  $DOWN$ . Wówczas prawdziwe jest przyporządkowanie:

$$l_A := x_{up}(0), \quad 1_A^- := x_{up}(1), \quad 1_A^+ := x_{down}(1), \\ p_A := x_{down}(0).$$

Po dodaniu funkcji stałej i równej 1 na przedziale  $[1_A^-, 1_A^+]$  otrzymamy  $UP$  i  $DOWN$  z jednym przedziałem (rys2, gdzie:  $\mu_{down} = x_{down}$ ,  $\mu_{up} = x_{up}$ ), który możemy potraktować jako nośnik. Wówczas funkcję przynależności  $\mu_A(x)$  zbioru rozmytego określonego na  $R$  definiują równania:

$$\mu_A(x) = 0 \text{ dla } x \notin [l_A, p_A] \\ \mu_A(x) = x_{up}^{-1}(x) \text{ dla } x \in UP$$

$$\mu_A(x) = x_{down}^{-1}(x) \text{ dla } x \in DOWN$$

W przypadku takiego zdefiniowania zbioru rozmytego otrzymamy dodatkową własność, która nazywana jest **skierowaniem**. Nośnikiem jest przedział  $UP \cup [1_A^+, 1_A^-] \cup DOWN$ .

Wartościami granic części up i down są:

$$\mu_A(l_A) = 0$$

$$\mu_A(1_A^-) = 1$$

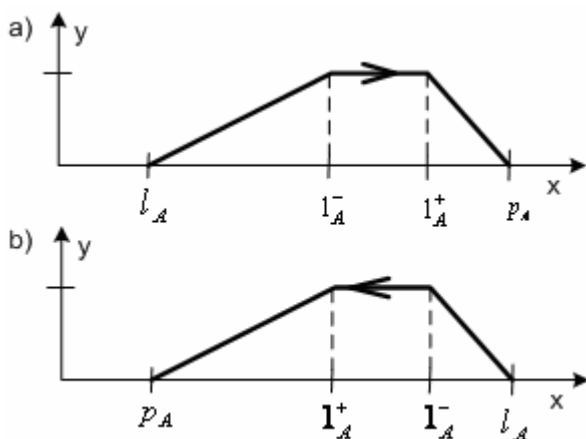
$$\mu_A(1_A^+) = 1$$

$$\mu_A(p_A) = 0$$

Ogólnie można przyjąć, że skierowane liczby rozmyte mają postać, trapezoidalną. Każdą z nich można opisać za pomocą czterech liczb rzeczywistych:

$$A = (l_A \quad 1_A^- \quad 1_A^+ \quad p_A).$$

Poniższe rysunki przedstawiają przykłady skierowanych liczb rozmytych wraz z punktami charakterystycznymi.



Rysunek 3. Liczba rozmyta skierowana a) dodatnio B) ujemnie.

**Funkcją przynależności skierowanej liczby rozmytej** A nazywamy funkcję  $\mu_A: R \rightarrow [0,1]$  określoną dla  $x \in R$  następująco:

$$\text{jeśli } x \notin \sup p_A \quad \Rightarrow \mu_A(x) = 0$$

$$x \in (1_A^-, 1_A^+) \quad \Rightarrow \mu_A(x) = 1$$

$$x \in \sup p_A \wedge x \notin (1_A^-, 1_A^+)$$

$$\Rightarrow \mu_A(x) = \max(f_A^{-1}(x), g_A^{-1}(x)).$$

Powyższą funkcję przynależności wykorzystać można w regułach sterowania w sposób podobny do wykorzystania przynależności klasycznych liczb rozmytych. Wszystkie wielkości występujące w sterowaniu rozmytym opisują wybrany element rzeczywistości. Proces określania tej wartości nazywany jest **obserwacją rozmytą**.

Do graficznego przedstawienia skierowanych liczb rozmytych wykorzystuje się **krzywe przynależności**. Wprowadzenie ich wymaga znajomości pojęcia zorientowanych liczb rozmytych.

**Zorientowaną liczbę rozmytą (ang. oriented fuzzy number)** rozumiemy jako trójkę  $A = (R, \mu_A, s_A)$ , gdzie  $\mu_A: R \rightarrow [0,1]$  jest funkcją przynależności tej liczby, zaś  $s_A \in \{-1, 0, 1\}$  oznacza skierowanie.

Ze względu na **orientację** liczby rozmyte można podzielić na dwie grupy:

OFN o orientacji pozytywnej – o kierunku zgodnym z kierunkiem narastania osi OX

OFN o orientacji negatywnej – o kierunku przeciwnym do kierunku narastania osi OX

Graficzną reprezentacją zorientowanej liczby rozmytej A jest para  $A = (C_A, s_A)$ . Przy czym  $C_A \subseteq R^2$  jest krzywa opisaną w sposób następujący:

$$C_A = ([l_A, p_A] \times \{0\}) \cup [1_A^-, 1_A^+] \times \{1\} \cup up_A \cup down_A$$

gdzie:  $up_A$  i  $down_A \subseteq R^2$  i są wykresami monotonicznych funkcji;

$s_A$  oznacza skierowanie;

zaś  $C_A$  to krzywa przynależności.

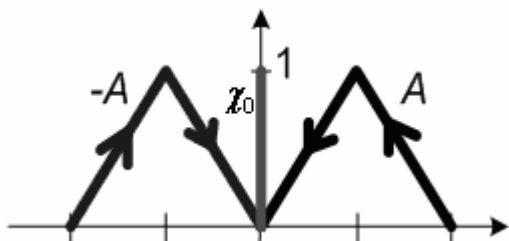
Funkcje  $f_A, g_A$  odpowiadają odpowiednio częściom  $up_A$ ,  $down_A \subseteq R^2$  tak, że:

$$up_A = \{(f_A(y), y) : y \in [0,1]\}$$

$$down_A = \{(g_A(y), y) : y \in [0,1]\}$$

Orientacja odpowiada kolejności wykresów  $f_A$  i  $g_A$ .

Graficzną interpretację dwóch liczb rozmytych przeciwnych i liczby rzeczywistej  $\chi_0$  przedstawia poniższy rysunek.



Rysunek 4. Liczby przeciwne oraz liczba rzeczywista

Liczby przeciwne mają przeciwne skierowanie. **Odwroceniem orientacji** skierowanej liczby rozmytej  $A$  nazywamy zamianę części *up* (funkcji  $f_A$ ) z częścią *down* (funkcją  $g_A$ ). Działanie to oznaczane jest następująco:

$$B = A |^- \Leftrightarrow g_B = f_A \wedge f_B = g_A$$

gdzie:

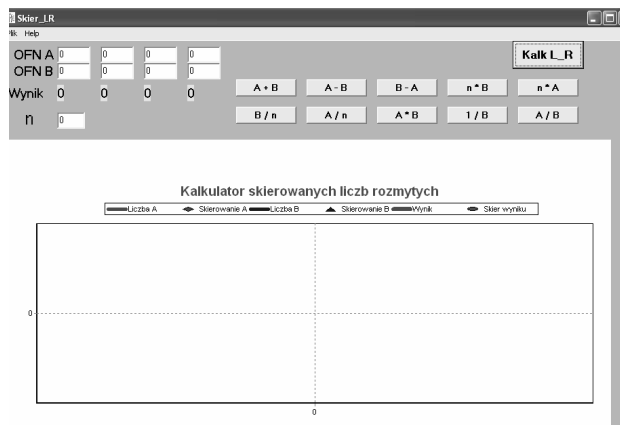
$A$  oznacza skierowaną liczbę rozmytą określoną przez parę funkcji  $(f_A, g_A)$ ,

$B$  jest wynikiem operacji odwracania orientacji OFN,  $|^-$  jest symbolem odwracania orientacji OFN.

Liczbę otrzymaną w ten sposób nazywamy liczbą odwróconą OFN lub liczbą o odwrotnej orientacji.

### 6. DZIAŁANIA NA SKIEROWANYCH LICZBACH ROZMYTYCH Z WYKORZYSTANIEM KALKULATORA PORÓWNAWCZEGO SKIER\_LR

Program *Skier\_LR* został stworzony w celu przedstawienia graficznej interpretacji, wykonywanych operacji na skierowanych liczbach rozmytych. Dodatkowo wyposażony jest w moduł *Kalk L-R*. Dzięki temu można porównać niektóre działania na tych liczbach. Zaimplementowany został w programie Delphi.



Rysunek 5. Okno główne programu

W oknie edycji tekstu wpisujemy wartości liczby rozmytej  $A$  i  $B$  oraz współczynnik  $n$ . Liczba OFN  $A$  ma postać  $(a1, 0), (a2, 1), (a3, 1), (a4, 0)$  a liczba  $B$   $(b1, 0), (b2, 1), (b3, 1), (b4, 0)$ . Po wybraniu działania zostają one wyświetlone wraz z wynikiem na wykresie.

```
begin
a1:=StrToFloat(edit1.Text);
a2:=StrToFloat(edit2.Text);
... ..
```

Wyczyszczenie liczby  $A$  na wykresie następuje po wykonaniu przez program poniższego kodu:

```
Series2.Clear;

W dalszej części programu punkty liczby  $A$  zostają połączone w podanej kolejności
series2.XValues.Order:=loNone;
Series2.AddXY(a1,0);
Series2.AddXY(a2,1);
Series2.AddXY(a3,1);
Series2.AddXY(a4,0);
... ..
```

Wykreślenie skierowania liczby  $A$  na wykresie:

```
Series5.AddXY(a1,0);
... ..
end;
```

Po dokonaniu wyboru działania wykonywane są programy, których fragmenty przedstawione są poniżej. Po wybranego zadania następuje wyświetlenie liczb  $A$  i  $B$ .

#### Suma liczb $A$ i $B$ .

Po wprowadzeniu danych program wykonuje następujące operacje:

```
c1:=a1 + b1;
c2:=a2 + b2;
c3:=a3 + b3;
```

c4:=a4 + b4;

Program wyświetla uzyskane wyniki:  
Label4.Caption:=FloatToStr(c1);  
... ..

Następuje wyczyszczenie wyniku poprzedniego i rysowanie wyniku operacji na wykresie oraz jego skierowania.

```
Series1.Clear;
Series1.XValues.Order:=loNone;
Series1.AddXY(c1,0);
Series1.AddXY(c2,1);
Series1.AddXY(c3,1);
Series1.AddXY(c4,0);
```

```
Series4.Clear;
Series4.AddXY(c1,0);
```

Podczas **odejmowania** program realizuje kod:

```
c1:=a1 - b1;
c2:=a2 - b2;
.....
```

Dalsza część programu jest taka sama jak przy dodawaniu.

Mnożenie przez stałą *n*:

```
c1:= a1 * n;
c2:= a2 * n;
... ..
```

Dzielenie skierowanej liczby rozmytej przez *n* zabezpieczone jest przed dzieleniem przez zero.

```
if n <> 0 then
begin
c1:= a1 / n;
c2:= a2 / n;
... ..
```

*W przeciwnym wypadku do powyższego działania następuje wyświetlenie komunikatu "Dzielenie przez zero!"*:

```
end else
Label9.Caption:='Dzielenie przez 0!';
end;
```

**Mnożenie OFN** wymaga wyznaczenia równań prostych, będących ramionami liczby rozmytej.

Wyznaczanie współczynnika kierunkowego prostych - ramion liczby rozmytej

```
if a2-a1 <> 0 then begin
upa11:=1/(a2-a1);
end else
upa11:=0 ;
```

```
if a1-a2 <> 0 then begin
upa12:=a1/(a1-a2);
end else
upa12:=0 ;
... ..
```

W dalszej części następuje wyznaczenie końców przedziałów iloczynu liczb rozmytych A\*B

```
c1:=a1*b1;
c2:=a2*b2;
... ..
```

Podział przydziału (0,1) na osi *y* i wyznaczenie poszczególnych punktów:

```
m1:=1/10;
m2:=m1*2;
... ..
```

W dalszej części wyznaczone zostają wartości argumentów dla ramion *up* liczby A i B dla wartości *m1*, *m2*. itd.

```
m1, m2...
p1:=m1*(a2-a1)+a1;
p2:=m2*(a2-a1)+a1;
... ..
o1:=m1*(b2-b1)+b1;
o2:=m2*(b2-b1)+b1;
... ..
```

Poniższy kod przedstawia wyznaczanie kolejnych punktów między przedziałem iloczynu A\*B oraz obliczanie dla nich wartości *y* (*fc1*, *fc2* itd.

```
fc1:=p1*o1;
fc2:=p2*o2;
... ..
```

Dalej wyznaczana jest wartość argumentów dla ramion *down* liczby A i B dla wartości *m1*, *m2*, itd.

```
p11:=m1*(a3-a4)+a4;
p22:=m2*(a3-a4)+a4;
... ..
o11:=m1*(b3-b4)+b4;
o22:=m2*(b3-b4)+b4;
... ..
```

Obliczanie wartości *y* dla kolejnych punktów między przydziałem iloczynu A\*B:

```
fc11:=p11*o11;
fc22:=p22*o22;
... ..
```

**Horaz skierowanych liczb rozmytych A/B**

Dzielenie to mnożenie danej liczby przez liczbę odwrotną do dzielnika. Procedura wyznaczania ramion, wartości argumentów, podziału przedziału (0,1) na osi y i ilustracji graficznej na wykresie jest taka sama jak przy mnożeniu. Końce przedziałów iloczynu OFN A i odwrotności OFN B i wyznacza następująca procedura:

```

c1:=a1*(1/b1);
c2:=a2*(1/b2);
... ..
m1:=1/10;
m2:=m1*2;
... ..

```

Program wyznacza wartości argumentów dla ramion up i down dla danych punktów, wyznacza kolejne punkty odwrotności i oblicza dla nich wartości.

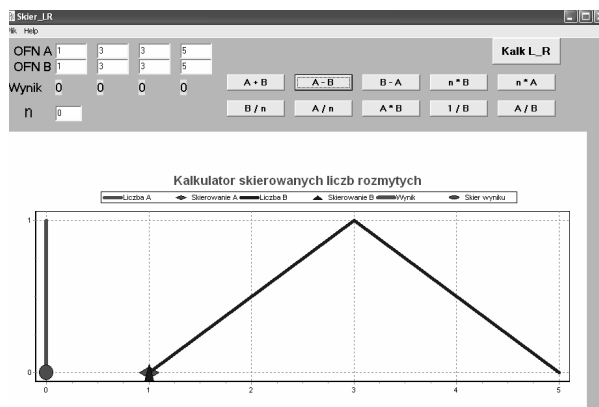
```

fc1:=1/o1;
... ..
n1:=p1*fc1;
... ..
p11:=m1*(a3-a4)+a4;
... ..
o11:=m1*(b3-b4)+b4;
... ..
fc11:=1/o11;
... ..
n11:=p11*fc11;

```

**Skier\_LR** wyposażony jest w dodatkowy moduł, który uruchamiany jest za pomocą przycisku **Kalk L-R**. Zawiera on procedury obliczania sumy, różnicy i iloczynu liczb postaci L-R. Dzięki temu można porównać niektóre wyniki otrzymane na liczbach L-R i OFN. Wprowadzona liczba rozmyta ma postać  $(m, \alpha, \beta)$  gdzie  $\alpha$  i  $\beta$  to rozrzuty lewo- i prawostronne.

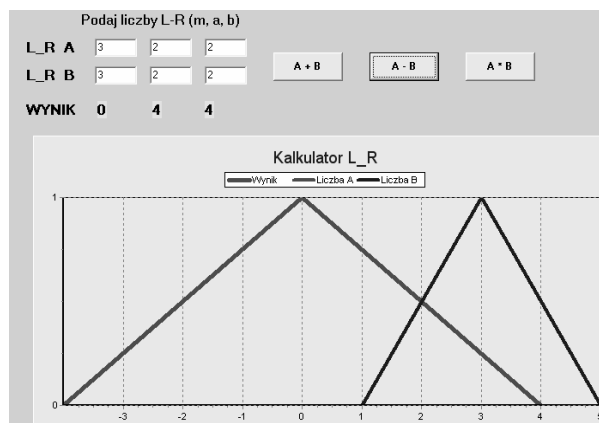
W wyniku przeprowadzonych testów za pomocą kalkulatora porównawczego można stwierdzić, że działania wykonywane na skierowanych liczbach rozmytych są w wielu dokładniejsze niż wykonywane na liczbach L-R gdyż nie zwiększa się tak bardzo ich nośnik. Odejmowanie liczb typu: 3-3, daje inny wynik w przypadku liczb LR a inny w przypadku OFN.



Rysunek 6. Wynik odejmowania 3-3 na liczbach OFN

Wynikiem tego działania jest zero rzeczywiste. Nośnik jest równy zero.

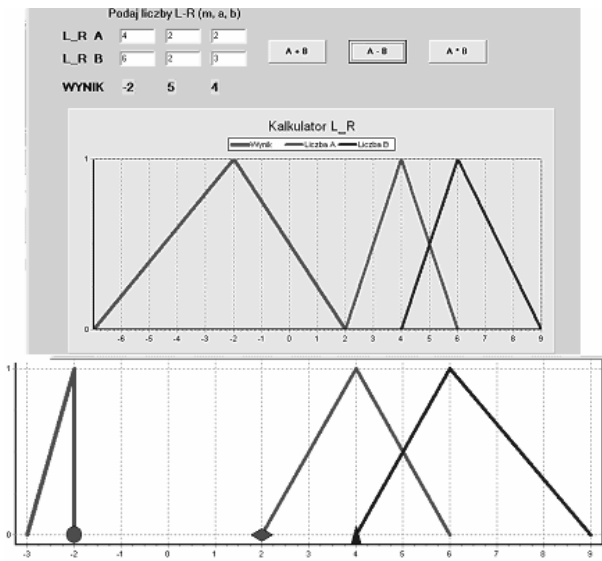
Poniższy rysunek przedstawia to działanie wykonane w programie **Kalk\_LR**.



Rysunek 7. Wynik odejmowania 3-3 na liczbach L-R

W pierwszym przypadku otrzymaliśmy zero rzeczywiste. Drugi wynik (działanie na L-R) to zero rozmyte.

Porównując wynik odejmowanie dwóch liczb postaci L-R przedstawione na rysunku nr 8 z wynikiem otrzymanym na przedstawionym tutaj kalkulatorze dojdziemy również do takiego samego wniosku jak poprzednio: w wyniku odejmowanie OFN zmniejsza się rozmycie.



Rysunek 8. Porównanie różnicy takich samych liczb w postaci L-R i OFN

Jak łatwo można zauważyć nośnik liczby otrzymanej wyniku w przypadku skierowanych liczb rozmytych jest mniejszy.

Każde działanie wykonywane na liczbach L-R niezależnie od tego czy jest to dodawanie czy odejmowanie zwiększa nośnik, czyli obszary nieprecyzyjności. Zatem wykonanie kilku działań na tych liczbach może spowodować takie rozmycie, że otrzymana wielkość nie będzie użyteczna.

Rozwiązanie równania typu  $A + X = C$  za pomocą reprezentacji L-R staje się niemożliwe do wykonania za pomocą działań, gdyż  $X + A + (-A) \neq 0$  oraz  $X * A * A^{-1} \neq X$ .

Każde działanie odwrotne na liczbach L-R spowoduje powiększenie nośnika. Nie da się tego rozwiązać metodą obliczeniową. Jedyną metodą prób i błędów.

Inaczej jest w wyniku działań na skierowanych liczbach rozmytych. Rozwiązanie takiego równania jest wykonywalne metodą obliczeniową.

*Przykład*

Wyznaczyć liczbę X spełniającą równanie  $A + X = C$

$$A(1,1,2) + X = C(5,2,3)$$

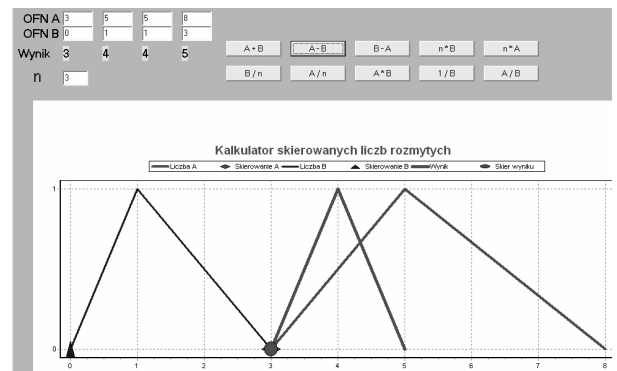
Rozwiązujemy to metodą obliczeniową.

$$A[0,1,1,3] + X = C[3,5,5,8]$$

$$X = C - A$$

$$X = [3,4,4,5]$$

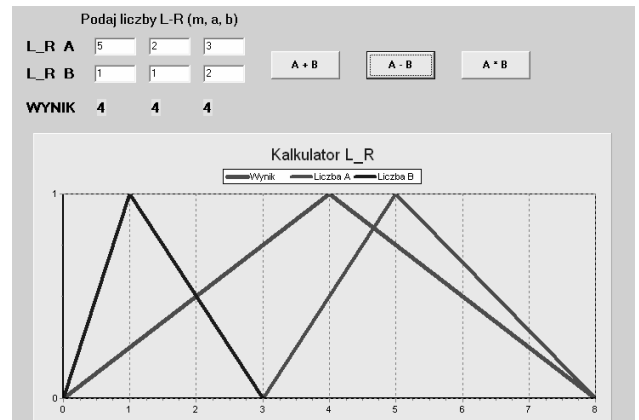
Przedstawiamy te liczby w postaci OFN i wykonujemy obliczenia za pomocą kalkulatora *Skier\_LR*.



Rysunek 9. Rozwiązanie równania  $A + X = C$  za pomocą odejmowania OFN

W wyniku obliczeń na liczbach L-R uzyskuje się wynik (4,4,4). Jednakże w wyniku sprawdzenia za pomocą dodawania ( $A + X$ ), uzyskamy wynik różny od C. Wynik prawidłowy (4,1,1) możemy uzyskać jedynie metodą prób i błędów. Jest to kłopotliwe i nie zawsze możliwe do wykonania.

Poniższy rysunek przedstawia rozwiązanie tego samego działania dla liczb L-R, wyznaczony za pomocą modułu *Kalk L-R*.



Rysunek 10. Rozwiązanie równania za pomocą liczb L-R

Sprawdzając poprawność rozwiązania tego równania w obydwu przypadkach, za pomocą liczb L-R otrzymujemy inny wynik zupełnie odmienny od początkowych liczb. Wynik otrzymany podczas sprawdzenia poprawności obliczenia jest zupełnie inny niż liczba wzięta początkowo.

Na liczbach OFN można stosować tak jak na liczbach rzeczywistych prawo rozdzielności mnożenia względem dodawania (odejmowania).

*Przykład*

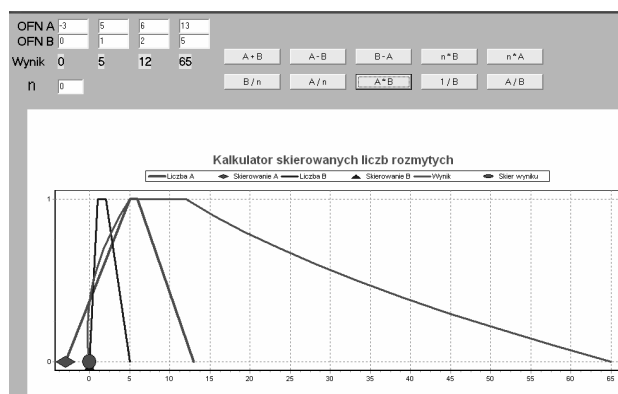
Dane są trzy skierowane liczby rozmyte A(-1,2,2,6), B (-2,3,4,7), C(0,1,2,5).

Sprawdzić prawo rozdzielności mnożenia względem dodawania

$$(A+B)*C = A*C + B*C$$

$$(A+B)*C = (0,5,12,65)$$

$$A*C + B*C = (0,5,12,65)$$



Rysunek 11 Wynik  $(A+B)*C$

## 7. WNIOSKI

Operacje wykonywane na skierowanych liczbach rozmytych są dokładniejsze niż działania wykonywane na klasycznych liczbach rozmytych. Wyniki działań wykonywanych na nich są takie same jak otrzymane na liczbach rzeczywistych. Wykonywanie kilku działań nie w każdym przypadku powoduje duży wzrost nośnika. Inaczej jest z liczbami rozmytymi, gdzie w wyniku kilku operacji otrzymujemy liczby „zbyt szerokie”. Nieskończenie mały nośnik interpretowany jest jako liczba rzeczywista i dzięki temu można na liczbach OFN stosować prawo łączności i przemienności mnożenia względem dodawania.

Możliwość wykonywania na nich wnioskowania wstecz daje szansę na odtworzenie danych wejściowych poprzez rozwiązanie równania.

Skierowane liczby rozmyte charakteryzuje łatwość i dokładność obliczeniowa. Przykładem może być mnożenie gdzie jednakowa procedura jest dla wszystkich *ordered fuzzy numbers*. Natomiast mnożenie liczb typu L-R jest inne dla dwóch liczb dodatnich, inne dla ujemnych, jeszcze inna procedura wykonuje mnożenie liczb o nieokreślonych znakach i różna dla zer rozmytych.

Opracowany kalkulator porównawczy *Skier LR* wykonuje działania na liczbach OFN i L-R dając możliwość porównania wyników tych działań i sformułowania odpowiednich wniosków.

## 8. KIERUNKI DALSZYCH BADAŃ

W ostatnich latach nastąpił znaczący rozwój logiki rozmytej, której idea jest oparta na podstawach matematycznych. Dlatego też niektóre środowiska programistyczne wyposażone są w moduły przeznaczone na potrzeby sztucznej inteligencji. Przykładem może być Matlab wyposażony w Fuzzy Logic Toolbox, Statistica z Data Miner czy też Mathematica z modułem Fuzzy Logic. Zawierają one odpowiednie narzędzia do tworzenia i modyfikacji systemów opartych na logice rozmytej. Dzięki nim można nie tylko tworzyć, ale i testować złożone systemy rozmyte. Wykorzystanie w nich skierowanych liczb rozmytych stwarza możliwość uzyskania dokładnych danych na podstawie nieprecyzyjnych informacji.

Dzięki liczbom OFN można zwiększyć dokładność nieprecyzyjnych danych. Takie zjawiska jak wzrost czy spadek danej wielkości można nazwać obserwacją rozmytą. Monotoniczność zjawiska może wpłynąć na skierowanie liczby ilustrującej go. Zatem wprowadzenie OFN stwarza nowe możliwości dla projektowania systemów o dużej dynamice. Ujęcie odpowiednie danego tematu przez projektanta daje szansę zastosowania skierowania dla danej właściwości i przedstawieniu go w pojęciu skierowanym. Skoro zaś można wskazać jego skierowanie daje to nowe możliwości dla rozwoju sterowania rozmytego, wytycza nowe drogi badań w dziedzinie logiki rozmytej.

Z każdym rokiem wydawanych jest coraz więcej książek oraz artykułów poświęconych jej zagadnieniom i zastosowaniu. Poszerzenie jej o teorię skierowanych liczb rozmytych pozwala na lepsze wykorzystanie nieprecyzyjnych działań. Prosta algorytmizacja *ordered fuzzy numbers* daje możliwość ich zastosowania w nowym modelu sterowania, inspiruje do poszukiwania nowych rozwiązań.

## Literatura

1. Kacprzak D. „Analiza modelu Leontiewa z użyciem skierowanych liczb rozmytych”, referat wygłoszony na II Konferencja Technologie Eksploracji i Reprezentacji Wiedzy, Białystok 2007  
[http://wi.pb.edu.pl/oldwi/terw2007/resources/articles/kacprzak\\_terw2007.pdf](http://wi.pb.edu.pl/oldwi/terw2007/resources/articles/kacprzak_terw2007.pdf), styczeń 2008.
2. Kosiński W. „On fuzzy number Calculus”, Int. J. Appl. Math. Comput. Sci., 2006, Vol. 16, No 1, 51-57.
3. Kosiński W. „Calculation and reasoning with ordered fuzzy numbers”; EUSFLAT-LFA 2005 Joint Conference;  
[http://www.eusflat.org/publications/proceedings/EUSFLAT-LFA\\_2005](http://www.eusflat.org/publications/proceedings/EUSFLAT-LFA_2005)
4. Kosiński W., Koleśnik R., Prokopowicz P., Frischmuth K., „On Algebra of Ordered Fuzzy

- Numbers”, *Soft Computing Foundations and Theoretical Aspects*, J.Kacprzyk i in. str.291-302, Warszawa 2004.
5. Kosiński W., Prokopowicz P., „Algebra liczb rozmytych”, str.37-63, *Matematyka Stosowana* nr 5/2004.
  6. Kosiński W., Prokopowicz P., Ślęzak D., „On Algebraic Operations on Fuzzy Numbers”, *Intelligent Information Processing and Web Mining: proceedings of the International IIS:IIPWM’03 Conference held In Zakopane, Poland, June 2-5, 2003*.
  7. Kosiński W., Słysz P. „Fuzzy numbers and their quotient space with algebraic operations, *Bull. Polish Acad. Sci.Ser. Tech. Sci.*41, str. 285-295, 1993 r.
  8. Miś B. „Rozmyty świat”, *„Wiedza i życie”* nr 2/1997 No. 1, 125–140
  9. Piotrowski T. „Wykorzystanie logiki rozmytej w diagnostyce transformatorów metodami DGA na przykładzie normy IEC – 60599”, *Instytut Elektroenergetyki, Łódź 2005*.
  10. Prokopowicz P. „Algorytmizacja działań na liczbach rozmytych i jej zastosowanie”; *rozprawa doktorska, Warszawa, kwiecień 2005*.
  11. Smejda A. „Application of the Fuzzy Set Theory to Assessment of the Entity’s Ability to Continue as a Going Concern”, *„Bank i kredyt”* lipiec 2006.
  12. Trzęsicki K. „Wkład logików polskich w światową informatykę”, referat, *Konferencja Technologie Eksploracji i Reprezentacji wiedzy, Białystok 2006*.



# MODELE RYWALIZACJI I WALKI W RÓWNANIACH LANCHESTERA

**Marcin Bajek**

Uniwersytet Kazimierza Wielkiego  
Instytut Techniki  
II rok MU Edukacja Techniczno-Informatyczna  
ul. Chodkiewicza 30, 85-064 Bydgoszcz  
e-mail: bayek@interia.pl

**Streszczenie:** Celem opracowania jest przedstawienie stworzonego przez Lanchestera modelu walki na dowolnym przykładzie. W artykule zamieszczone są informacje na temat samego modelu walki, prezentowany jest ogólny układ równań Lanchestera oraz układ klasyczny i najpopularniejszy. Do komputerowej symulacji wykorzystane będą aplikacje w Excel-u oraz C++ Builder. Układ równań jest rozwiązany metodami numerycznymi z wykorzystaniem czteropoziomowego algorytmu Runge–Kutty.

**Słowa kluczowe:** Lanchester, układ równań Lanchestera, metoda Runge–Kutty

## Competition and combat models in Lanchester equations

**Abstrakt:** The aim of this study is to present the combat model created by Lanchester using some example. The paper includes information on the combat model itself. The author presents Lanchester's general set of equations as well as classical and most popular sets. The computer simulation uses applications created in Excel and C++ Builder. The set of equations is solved by means of numerical methods using four-level Runge–Kutty algorithm.

**Keywords:** Manchester, Lanchester's set of equations, the Runge–Kutty method

### 1. RÓWNANIA LANCHESTERA

Frederick William Lanchester sformułował kilka prostych równań różniczkowych opisujących modele walki. Okazały się one bardzo ciekawym, bezkrwawym narzędziem zdolnym do przewidywania czasu zakończenia bitwy czy zwycięzcy starcia. Równania te sprawdzały się przy analizach starć wojennych od czasów antycznych po czasy okresu boni palnej. Rozszerzone zostały też zastosowane do innych operacji, np. w przemyśle, giełdach papierów wartościowych. Każda modyfikacja równań wiązała się ze wzrostem liczby zmiennych i dodawanych kolejnych współczynników. Ogólny model Lanchestera w żaden sposób nie nadaje się do bezpośredniego zastosowania we współczesnym polu np. działań zbrojnych.

Ogólny model Lanchestera wygląda następująco [1]:

$$\begin{cases} \frac{dR(t)}{dt} = -bB(t)^{c_1} \cdot R(t)^{c_2} \\ \frac{dB(t)}{dt} = -rR(t)^{c_3} \cdot B(t)^{c_4} \end{cases}$$

(1)

gdzie:

$r, b$  – liczby rzeczywiste z przedziału  $(0,1)$ , tak zwane współczynniki efektywności Lanchestera lub

$$\begin{cases} \frac{dR(t)}{dt} = -bB(t) \\ \frac{dB(t)}{dt} = -rR(t) \end{cases}$$

współczynniki waleczności. Interpretuje się je często jako prawdopodobieństwa strat każdej z walczących stron w ciągu umówionej jednostki czasu walki,

$\frac{dR(t)}{dt}$  oraz  $\frac{dB(t)}{dt}$  – straty stron,

$c_1, \dots, c_4$  – wykładniki potęg przy  $B(t)$  i  $R(t)$ , mogą przyjmować wartości liczbowe 0 lub 1.

Najprostszy z wariantów ma postać [3]:

(2)

Powyższy układ jest najpopularniejszym modelem i w dalszej części to właśnie on będzie rozpatrywany.

**9. NUMERYCZNE METODY ROZWIĄZYWANIA RÓWNAŃ RÓŻNICZKOWYCH ZWYCZAJNYCH**

Podstawowe równanie różniczkowe pierwszego rzędu ma postać:

$$\frac{dy}{dx} = f(x, y) \quad (3)$$

gdzie  $f(x, y)$  jest znaną funkcją.

Przyjmując, że  $dy = y_{n+1} - y_n$  oraz  $dx = h$  otrzymany jest prosty algorytm numeryczny:

$$\frac{y_{n+1} - y_n}{h} = f(x_n, y_n) \quad (4)$$

$$y_{n+1} = y_n + f(x_n, y_n) \cdot h \quad (5)$$

$$x_n = x_0 + n \cdot h \quad (6)$$

Znając warunki początkowe  $x_0, y_0$  oraz zakładając wartość kroku  $h$  można znaleźć zależność  $y(x)$  dla  $x \in \langle x_0, x_n \rangle$

Równanie (5) przedstawia krok następny w jednokrokowym schemacie Eulera.

W przypadku gdy w równaniach różniczkowych występują pochodne rzędu wyższego niż pierwszy to takie równanie jest przekształcane do układu równań rzędu pierwszego. Dzięki temu można do rozwiązania zastosować metody numeryczne.

Jedną z najbardziej rozpowszechnionych i popularnych jest czterokrokowa metoda Rungego – Kutty. Zalicza się ją do bardzo wygodnych w programowaniu ze względu na swoją prostotę. Struktura klasycznej metody Rungego – Kutty dla układu równań wygląda następująco [2]:

$$k_1 = hf(x_n, y_n)$$

$$m_1 = hf(x_n, y_n)$$

$$k_2 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right)$$

$$m_2 = hf\left(x_n + \frac{h}{2}, y_n + \frac{m_1}{2}\right)$$

$$k_3 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right)$$

$$m_3 = hf\left(x_n + \frac{h}{2}, y_n + \frac{m_2}{2}\right)$$

$$k_4 = hf(x_n + h, y_n + k_3)$$

$$m_4 = hf(x_n + h, y_n + m_3)$$

$$x_{n+1} = x_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (7)$$

$$y_{n+1} = y_n + \frac{1}{6}(m_1 + 2m_2 + 2m_3 + m_4)$$

**10. IMPLEMENTACJA KOMPUTEROWA MODELU**

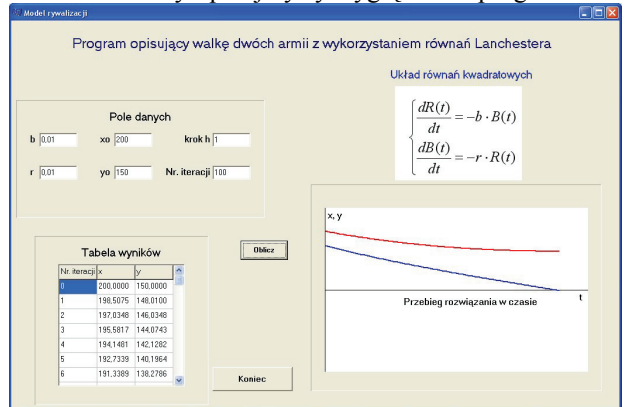
Implementacja w Excel-u

Środowisko Excel świetnie nadaje się porozwiązywania zadań metodami numerycznymi ze względu na łatwość implementacji równań. Innym ale bardzo ważnym atutem Excel-a jest prostota w tworzeniu niezbędnych wykresów.

Rysunek. 1 Implementacja układu równań Lanchestera

Implementacja w Borland C++ Builder

Zaletami programu są na pewno: działanie aplikacji bez potrzeby instalowania samego programu czyli istnieje możliwość prezentacji na każdym stanowisku bez potrzeby wcześniejszych przygotowań. Inne zalety to: łatwość w implementacji metody Rungego – Kutty oraz brak arkusza czyli przejrzysty wygląd okna programu.

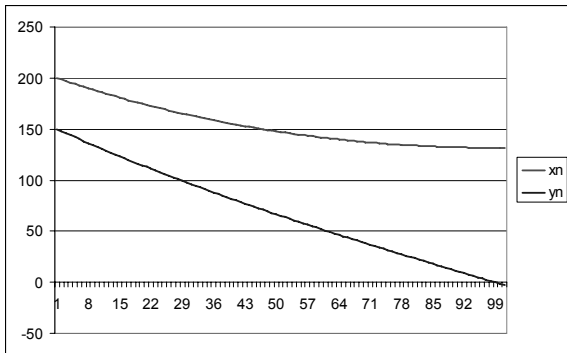


Rysunek. 2 Implementacja układu równań Lanchestera

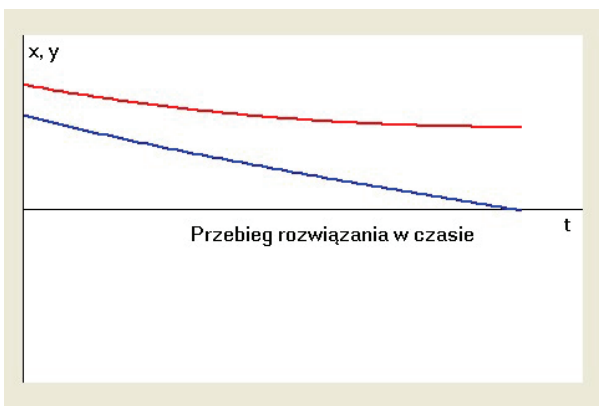
Przykład:

Dwie armie toczą ze sobą bitwę. Armia R (Red) liczy 200 tysięcy żołnierzy zaś armia B (Blue) liczy 150 tysięcy. Współczynnik efektywności obu armii wynosi 0,01. Obliczenia zostaną zakończone kiedy jedna ze stron poniesie porażkę czyli zginą wszyscy jej żołnierze [4].

Rozwiązanie:



**Rysunek. 3** Graficzne rozwiązanie układu równań w Excel-u. Na osi poziomej podano numer iteracji



**Rysunek. 4** Graficzne rozwiązanie układu równań w C++ Builder

Widać na obu wykresach jak niebieska linia osiąga wartość zerową, oznacza to, że porażkę poniosła armia B (Blue). Czynnikiem liczebności zdecydował o tym kto przegrał bo obie strony posiadały taki sam współczynnik waleczności. Czas trwania starcia wynosi 97 godzin lub np. minut – kwestia doboru jednostki czasu w modelowaniu jest dowolna. Po stronie R (Red) pozostanie 131 tysięcy żołnierzy więc klęska przeciwnika jest dość spora.

W prezentowanych aplikacjach można dokonywać różnych zmian, np. operować współczynnikami waleczności czy zmieniać krok  $h$ . Jeżeli liczebność walczących armii będzie taka sama (150 tys. żołnierzy) przy jednakowych współczynnikach waleczności (0,01) to zaobserwuje się ciekawe zjawisko. Starcie zakończone jest porażką obu armii, z tym że czas jej trwania zwiększy się aż pięciokrotnie. Inny przypadek to gdy przyjęte zostaną dane początkowe ale założone zostanie zwycięstwo armii B (Blue). Trzeba np. podwoić współczynnik waleczności dla tej właśnie armii. Wyniki starcia to oczywiście zaplanowane zwycięstwo, które zakończy się po ok. 130 godzinach (jednostkach) i 50 tys. żołnierzach przy życiu.

## 11. WNIOSKI I PODSUMOWANIE

Model walki Lanchestera był bardzo dobrze znany w pierwszej połowie XX wieku i stosowany praktycznie w każdej dziedzinie gdzie występowała rywalizacja. Obecnie przy tak zaawansowanej technologii informatycznej można w sposób bardzo łatwy każdy z modeli zaimplementować w dowolnej aplikacji i dokonywać symulacji oraz badań. Stało się to możliwe dzięki numerycznym metodom rozwiązywania układów równań różniczkowych.

Stworzone aplikacje w Excel-u oraz w C++ Builder umożliwiają rozwiązanie klasycznego modelu Lanchestera w sposób bardzo szybki i efektywny. Można jak pokazano to powyżej dokonywać wielu symulacji. Otrzymane wyniki w obu aplikacjach są identyczne więc nie ma znaczenia, który program będzie używany.

## Literatura

- 1 Spustek Henryk, „Model przewagi i jego implementacja komputerowa”, Akademska Oficyna Wydawnicza EXIT, Warszawa 2006
- 2 Degras Jean, „Praktyczne metody analizy numerycznej”, Wydawnictwo Naukowo Techniczne, Warszawa 1974.
- 3 Sadowski Witold, artykuł „Wojenne (PO)rachunki”, Wiedza i życie, 2001
- 4 <http://www.cs.uiowa.edu/~dsidran/ReadingsForResearch2.pdf>, pp.10-12



# OSCYLATOR RLC Z LOSOWYMI PARAMETRAMI REZYSTANCJI I INDUKCYJNOŚCI

Piotr Fajkowski

Uniwersytet Kazimierza Wielkiego  
Instytut Techniki  
II rok MU Edukacja Techniczno-Informatyczna  
ul. Chodkiewicza 30, 85-064 Bydgoszcz  
e-mail: piotras230@o2.pl

**Streszczenie:** W artykule prezentowany jest model szeregowego oscylatora RLC, który zaburzany jest losowymi parametrami rezystancji i indukcyjności. Do numerycznej symulacji układu został stworzony program napisany w języku C++. Program bada wpływ losowych zmian parametrów na oscylator RLC. Na podstawie tworzonych wykresów określamy stabilność pracy oscylatora. Głównym zadaniem jest znalezienie przybliżonej wartości losowego parametru, przy którym układ staje się nieprzewidywalny. Świadczyć o tym będą przecięcia trajektorii pojawiające się na portrecie fazowym układu RLC.

**Słowa kluczowe:** Oscylator, układ RLC, metody numeryczne, niestabilność

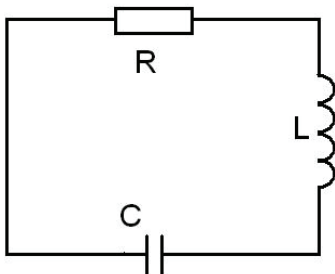
## RLC oscillator with random parameters of resistance and inductance

**Abstract:** The article presents the model of serial RLC oscillator that is disturbed by random parameters of resistance and inductance. Numerical simulation of the system has been realised by the program developed in C++ language. The program examines impact of random parameter changes on the RLC oscillator. Authors determine the stability of the oscillator operation on the grounds of created graphs. The main aim was to find approximate value of a random parameter for which the system becomes unpredictable. This will be indicated by trajectory intersections that occur in the phase portrait of the RLC system.

**Keywords:** Oscillator, RLC system, numerical analysis methods, instability

### 1. WSTĘP

Przedmiotem numerycznej analizy jest klasyczny układ RLC bez wymuszenia (rys. 1). Składa się on z szeregowo połączonego rezystora o rezystancji R, cewki o indukcyjności L i kondensatora o pojemności C.



Rys. 1 Klasyczny szeregowy układ RLC bez wymuszenia

Taki układ możemy opisać równaniem różniczkowym:

$$L \frac{d^2 i}{dt^2} + R \frac{di}{dt} + \frac{i}{C} = 0 \quad (1)$$

Równanie różniczkowe (1) opisuje zachowanie obwodu RLC, w którym drgania elektryczne układu są tłumione. Kondensator i cewka są źródłem drgań. Rezystor R wraz z przewodami łączącymi wszystkie elementy układu jest odpowiedzialny za tłumienie drgań wytwarzanych przez cewkę i kondensator. Wynika to z prawa Jula-Lenza zgodnie z którym energia układu jest tracona na elementach rezystancyjnych w postaci ciepła. Tak więc im większa rezystancja obwodu tym szybciej wzbudzone drgania będą gasły.

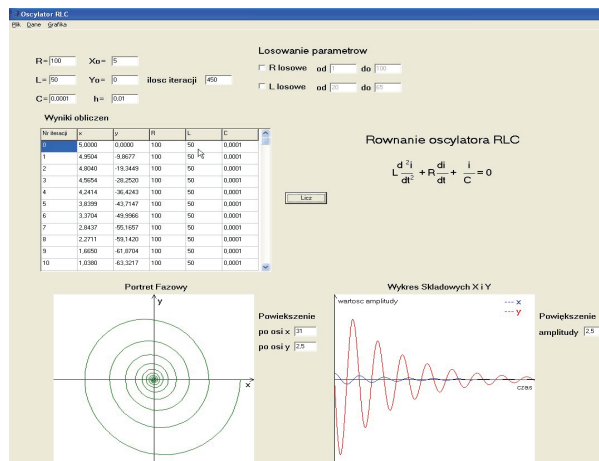
Równanie (1) jest punktem wyjścia do analizy numerycznej układu, należy je jednak odpowiednio przekształcić na równoważny układ równań (2):

$$\begin{cases} y' = -\frac{\left(R \cdot y + \frac{x}{C}\right)}{L} \\ x' = y = \frac{di}{dt} \end{cases} \quad (2)$$

Tak przygotowany układ równań można rozwiązać numerycznie metodą czteropoziomową Rungego-Kutty. Metoda ta bardzo dobrze nadaje się do rozwiązywania równań różniczkowych zwyczajnych ze zdefiniowanymi warunkami początkowymi oraz zapewnia odpowiednią dokładność rozwiązania. Schemat tworzenia równań według metody Rungego-Kutty jest omawiany w wielu opracowaniach [1].

Odpowiednio utworzony szereg równań zapisano w kodzie źródłowym programu, który rozwiązuje numerycznie równanie oscylatora RLC. Aplikacja została napisana w języku Builder C++. Program umożliwia badanie oscylatora bez losowych parametrów R i L jak też z losowymi parametrami. Głównym zadaniem programu jest ocena wpływu losowych parametrów na zachowanie się układu RLC. Ocena ta polegać będzie na wyszukiwaniu przecinających się linii na portrecie fazowym (bifurkacji) przy różnych wartościach losowych parametrów R i L. Układ będzie uznawany za pracujący stabilnie jeżeli na portrecie fazowym nie wystąpią żadne bifurkacje. Jeśli przecinające się linie będą obecne na wykresie fazowym to oscylator będzie wykazywał cechy układu chaotycznego.

Na rysunku 2 przedstawiono wygląd okna programu. Sama obsługa aplikacji nie sprawia kłopotów. Należy tylko właściwie dobrać parametry rezystancji R, indukcyjności L i pojemności C. Ustalić warunki początkowe z których ma startować rozwiązanie oraz odpowiednio ustalić krok h. Program będzie prowadził obliczenia z zadaną przez użytkownika ilością iteracji. Ilość iteracji możemy traktować jako ewolucję obwodu w czasie.



Rysunek 2 Wygląd programu rozwiązującego równanie różniczkowe oscylatora RLC

## 2. PRZEPROWADZONE BADANIA OSCYLATORA RLC Z LOSOWYMI PARAMETRAMI

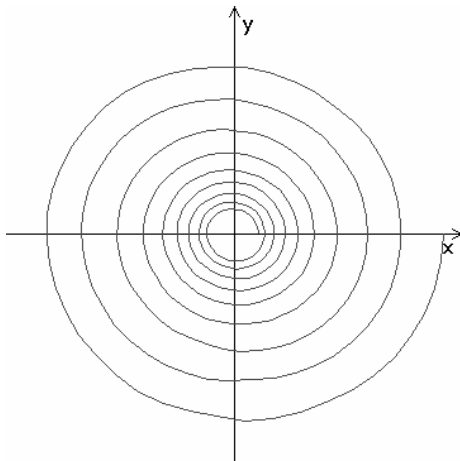
W tej części artykułu znajdują się przykładowe badania układu RLC z losowymi parametrami. Układ można modelować dla różnych wartości parametrów R, L i C, jednak zadaniem programu jest pokazanie jak wpływają one na oscylator. Prowadzone badania są dość unikalne, ponieważ losowe parametry są zmieniane w czasie pracy układu. Po każdej iteracji są losowane nowe wartości rezystancji i/lub indukcyjności z zadanego przedziału.

### 2.1. Oscylator RLC z losową rezystancją R

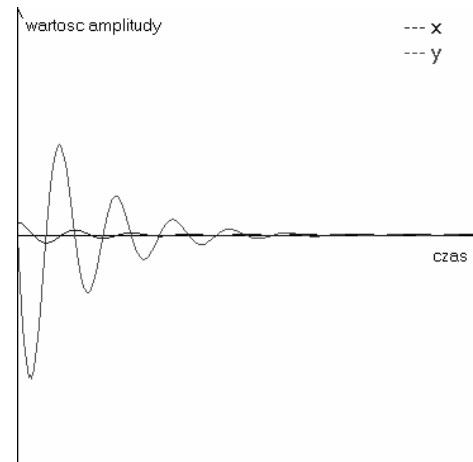
W symulacji układu z losową rezystancją przyjęto następujące stałe parametry dla oscylatora:

$$\begin{aligned} x_0 &= 5 & y_0 &= 0 \\ L &= 50\text{H} & C &= 0,0001\text{F} \\ h &= 0,01 & \text{ilość iteracji} &= 400 \end{aligned}$$

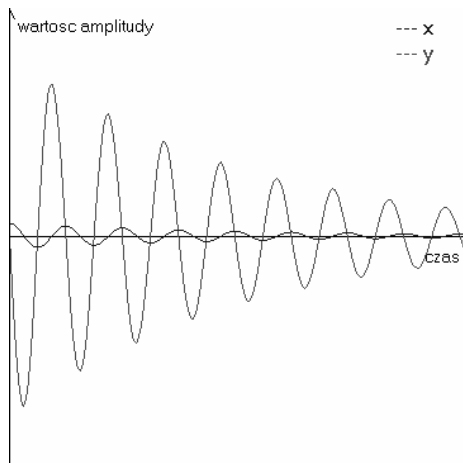
Symulacja z losową rezystancją R od 0 do 100Ω



Rys. 1 Oscylator RLC - portret fazowy

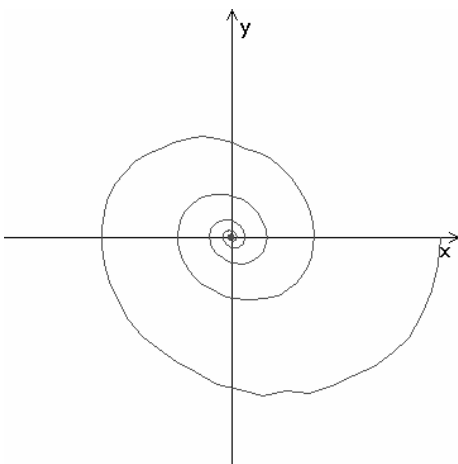


Rys. 4 Oscylator RLC - wykres drgań składowych x i y



Rys. 2 Oscylator RLC - wykres drgań składowych x i y

2) Symulacja z losową rezystancją od 50 do 350Ω



Rys. 3 Oscylator RLC - portret fazowy

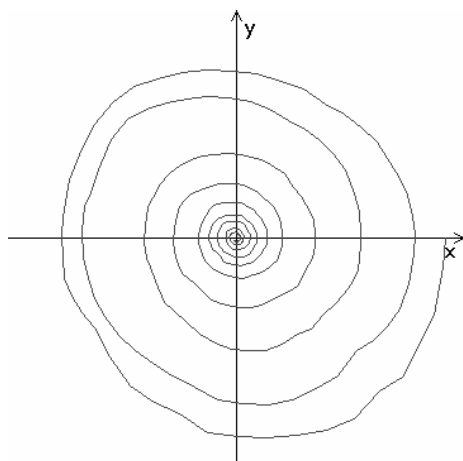
Przyglądając się powyższym rysunkom widzimy że losowa rezystancja w małym stopniu wpływa na nasz układ. Nie występują tu żadne szukane bifurkacje. Zaburzenie portretu fazowego rys. 3a przez losową rezystancję od 0 do 100Ω jest praktycznie niewidoczne. Układ pracuje w granicach stabilności, a składowe x i y mają charakter drgań gasnących. W symulacji kiedy rezystancję losowano z zakresu od 50 do 350Ω możemy zauważyć pewne zafalowania na portrecie fazowym (rys. 4a). Jest to spowodowane dużym zakresem losowania (300Ω), gdzie rezystancja zmienia się skokowo o dość duże wartości. Duży zakres losowania powoduje że drgania jeszcze szybciej gasną niż na rys. 3, ponieważ górna wartość rezystancji wynosi aż 350Ω. Rezystancja R nie zaburza w znaczący sposób pracy oscylatora RLC. Częstotliwość drgań własnych jest stała. Duża rozpiętość przedziału losowania rezystancji sprawia że układ wytraca swoją energię przez co drgania szybciej gasną.

2.2 Oscylator RLC z losową indukcyjnością L

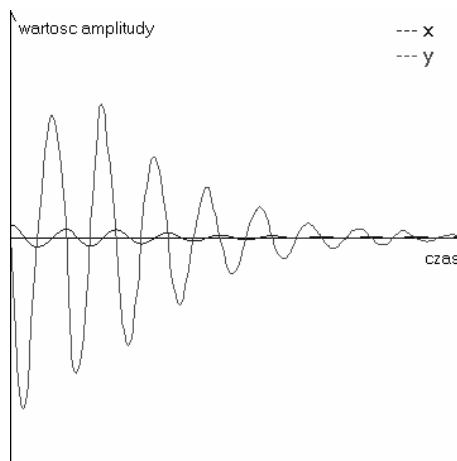
W symulacji układu z losową indukcyjnością przyjęto następujące stałe parametry:

$$\begin{aligned} x_0 &= 5 & y_0 &= 0 \\ R &= 100\Omega & C &= 0,0001F \\ h &= 0,01 & \text{ilość iteracji} &= 400 \end{aligned}$$

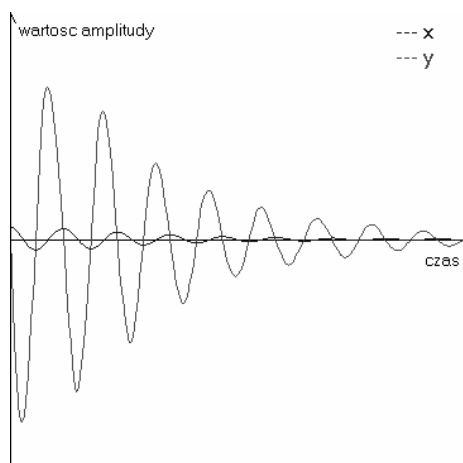
1) Symulacja z losową indukcyjnością L od 25 do 75H



Rys. 5 Oscylator RLC - portret fazowy

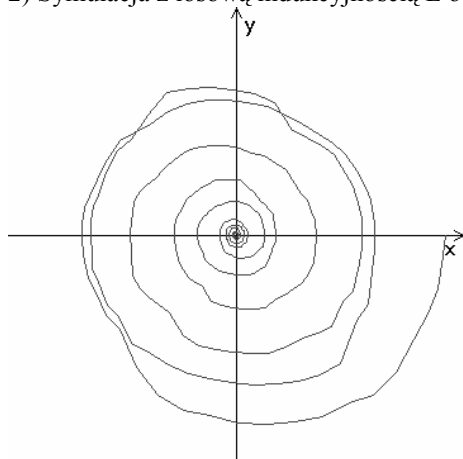


Rys. 8 Oscylator RLC - wykres drgań składowych x i y



Rys. 6 Oscylator RLC - wykres drgań składowych x i y

2) Symulacja z losową indukcyjnością L od 15 do 85H



Rys. 7 Oscylator RLC - portret fazowy

Przy zaburzaniu układu RLC losową indukcyjnością pojawiają się szukane bifurkacje (rys. 6a). Indukcyjność losowana z zakresu od 15 do 85H powoduje powstanie przecinających się linii na portrecie fazowym. Dla tak dobranych parametrów układu jest to umowna graniczna wartość przy której zaczynają się pojawiać bifurkacje. Na wykresie 6a widzimy ewolucję układu. Przecinające się linie wykresu fazowego nie pozwalają przewidzieć jaką trajektorię obierze układ. Staje się on nieprzewidywalny. Na wykresie drgań amplitud (rys. 6b) możemy dostrzec że drgania są bardzo chaotyczne, kolejno zwiększając i zmniejszając swoją amplitudę. Po przekroczeniu umownej granicy wystąpienia bifurkacji efekt zaburzania oscylatora nasila się i nie jesteśmy w stanie przewidzieć jego pracy. Charakter obwodu gwałtownie może się zmieniać łącząc w sobie drgania silnie gasnące, stałe, czy bardzo szybko rosnące. Już na rys. 5 widać że losowa indukcyjność w większym stopniu zaburza oscylator RLC niż losowa rezystancja. Wpływa ona znacząco na stabilność oscylatora.

### 3. PODSUMOWANIE

Przeprowadzone symulacje pozwalają stwierdzić że głównym czynnikiem powodującym powstawanie niestabilności na portrecie fazowym jest indukcyjność. Wiemy że częstotliwość drgań obwodu RLC zależy od iloczynu LC. Dlatego losowa indukcyjność i losowa pojemność (nie prezentowana w tym artykule) znacząco wpływają na pracę całego układu. To one są odpowiedzialne za bardzo silne zaburzenia oscylatora. To losowa indukcyjność powoduje że pracy układu nie jesteśmy w stanie w żaden sposób przewidzieć. Oscylator ewoluje zgodnie z losowością parametru L. Rezystancja natomiast tłumi drgania i im większy

przedział jej losowania zastosujemy tym szybciej będzie wytracana energia układu. Oczywiście prezentowane tu symulacje są przykładowe i pokazują pracę układu RLC opisanego równaniem różniczkowym (1). Równie dobrze możemy testować nasz oscylator dla innych wartości R, L, C oraz warunków początkowych. Jednak schemat pracy układu powinien być podobny do omawianego tutaj.

Temat losowo zmieniających się parametrów w czasie pracy układu jest dość unikalny, ponieważ w większości prac losowo zmieniane są parametry początkowe układów [2]. Przyglądając się bliżej badanemu problemowi może on częściowo symulować starzenie elementów układu. W wielu przypadkach dzieje się tak, że zanim nastąpi awaria urządzenia to jeden z jego elementów traci swoje pożądane właściwości i zaczyna zachowywać się nieprzewidywalnie. Prezentowany w tym artykule układ może właśnie takie zjawisko symulować poprzez zmieniające się w czasie wartości danego parametru.

#### **Literatura**

1. Jean Legras: Praktyczne metody analizy numerycznej, WNT Warszawa 1974
2. Janicki A., Izydorczyk A. : Komputerowe metody w modelowaniu stochastycznym, WNT Warszawa 2001.
3. Drgania w obwodach RLC i fale elektromagnetyczne, <http://www.mif.pg.gda.pl/kfze/wyklady/WM2rozdzial7.pdf>
4. Programowanie w Borland C++ Builder, strona internetowa: <http://cyfbar.republika.pl/>



# RÓWNANIE VAN DER POLA

**Paweł Jendykiewicz**

Uniwersytet Kazimierza Wielkiego  
Instytut Techniki  
II rok MU Edukacja Techniczno-Informatyczna  
ul. Chodkiewicza 30, 85-064 Bydgoszcz  
e-mail: pawel25pl@wp.pl

**Streszczenie:** W artykule prezentowana jest metoda badania stabilności obwodu elektrycznego opisanego za pomocą równania różniczkowego. Aby opisać tą metodę wykorzystano oscylator van der Pola. Badania układu opisanego równaniem van der Pola z losowym współczynnikiem tłumienia przeprowadzono za pomocą programu vdP.exe. Na podstawie wizualizacji wyników rozwiązania równania vdP określamy przybliżoną wartość współczynnika tłumienia, przy którym układ traci stabilność. Wykorzystaną metodą numeryczną do obliczeń równania różniczkowego jest metoda Runge-Kutty według schematu cztero poziomowego.

**Słowa kluczowe:** Bifurkacja, Równanie van der Pola, metoda Runge-Kutty.

## Van der Pol equation

**Abstract:** This paper presents the electric circuit stability test method analysed using differential equation. Authors described this method using van der Pol oscillator. Research of the system analysed using van der Pol equation with random damping coefficient has been performed by means of the vdP.exe program. Authors determine approximate value of the damping coefficient for which the system becomes unstable on the grounds of the vdP equation visualisation results. The numerical analysis method used to calculate the differential equation is Runge-Kutty method with a four-level pattern.

**Keywords:** Bifurcation, Van der Pol equation, the Runge-Kutty metod

### 1. WSTĘP

Możliwość jednoznacznego określenia ewolucji w czasie interesującego nas procesu, istnieje gdy dysponujemy wiedzą dotyczącą zależności funkcyjnych między funkcją opisującą ten proces i jej pochodną (lub jej pochodnymi) oraz dodatkowo należy dysponować warunkami początkowymi. Zależność funkcyjna pomiędzy nieznaną funkcją a jej pochodną nosi nazwę równania różniczkowego.

Obecnie bardzo trudno wyobrazić sobie rozwój wielu dziedzin nauki bez znajomości teorii równań różniczkowych. Istnieje wiele kierunków rozwoju współczesnych teorii równań różniczkowych oraz różne metody ich nauczania w zależności od potrzeb uwarunkowanych kręgiem odbiorców.

Celem artykułu jest przedstawienie metody i określenie stabilności układu opisanego za pomocą równania van

der Pola (1). Przy czym wartość parametr  $\mu$  – czyli współczynnik tłumienia będzie liczbą losową. Warunki początkowe natomiast będą stałe.

### 2. RÓWNANIE VAN DER POLA

Równanie van der Pola (oscylator van der Pola) opisane jest następującym modelem matematycznym

$$\frac{d^2 x}{dt^2} + \mu \cdot (1 - x^2) \cdot \frac{dx}{dt} + x = 0 \quad (1)$$

$$\mu \cdot (1 - x^2) \cdot \frac{dx}{dt} \quad (2)$$

nieliniowa funkcja tłumienia,  
 $\mu$  – współczynnik tłumienia.

Równanie (1) jest to równanie wykorzystywane do analizy obwodu elektrycznego, w którym zwykły rezystor został zastąpiony elementem o nieliniowej

charakterystyce prądowo-napięciowej. Elementami o takiej charakterystyce są diody tunelowe.

Równanie (1) ma dokładnie jedno asymptotycznie stabilne rozwiązanie okresowe. Fakt ten jest podstawą wykorzystania diod tunelowych do konstrukcji tzw. generatorów samowzbudnych, niezbędnych w wielu zastosowaniach elektroniki (generowanie i odbiór sygnałów).

Szczegółowe wyprowadzenia równania vdP oraz zastosowanie równań różniczkowych w teorii obwodów elektrycznych możemy znaleźć w wielu opracowaniach. [1]

W ostatnich latach oscylator stosowany jest również do modelowania dynamiki serca.

Jedną z cech oscylatorów relaksacyjnych jest umiejętność dopasowywania swojej częstości do częstości zewnętrznego układu wymuszającego [2]. Ta cecha powoduje, że oscylatory relaksacyjne nadają się do modelowania układów, w których istotne jest generowanie odpowiedzi na pobudzenie, o dopasowującej się częstości i stałej amplitudzie. Tą własność wykorzystali van der Pol i van der Mark tworząc pierwszy model serca oparty na oscylatorach relaksacyjnych (van der Pola) [3].

### 3. NUMERYCZNE ROZWIĄZANIE RÓWNIANIA VAN DER POLA.

W rozlicznych opracowaniach [4],[5] można znaleźć algorytmy numerycznego rozwiązania równania różniczkowego lub układu równań. Prosty schemat drugiego rzędu przedstawia następujący schemat:

$$y_{n+1} = y_n + \frac{1}{2}(k_1 + k_2)$$

$$k_1 = h \cdot f(t_n, y_n), \quad k_2 = h \cdot f(t_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1)$$

$h$  - jest krokiem (przyrostem) całkowania zmiennej niezależnej  $t$ , natomiast  $k_1$  i  $k_2$  są to wyrażenia pomocnicze.

Bardziej dokładną metodą stosowaną przy całkowaniu równań różniczkowych jest metoda Runge-Kutty czwartego rzędu.

$$y_{n+1} = y_n + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} + 0(h^5) =$$

$$y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) + 0(h^5)$$

$$k_1 = h \cdot f(t_n, y_n), \quad k_2 = h \cdot f(t_n + \frac{h}{2}, y_n + \frac{k_1}{2})$$

$$k_3 = h \cdot f(t_n + \frac{h}{2}, y_n + \frac{k_2}{2}), \quad k_4 = h \cdot f(t_n + h, y_n + k_3)$$

Metoda w porównaniu do poprzedniej ma większą dokładność.

Rozwiązać równanie różniczkowe oznacza znaleźć wszystkie funkcje spełniające dane równanie różniczkowe, tzw. rozwiązanie ogólne. Funkcji tych jest zazwyczaj nieskończenie wiele, a różnią się zazwyczaj wielkościami jednego parametru.

Aby rozwiązać równanie (1), posłużymy się metodą, która pozwoli znaleźć szczególne rozwiązanie równania. Spełniając pewne warunki zwane brzegowymi lub początkowymi.

Na początku musimy nasze równanie różniczkowe (1) sprowadzić do postaci układu równań różniczkowych pierwszego rzędu.

$$\begin{cases} x' = y \\ y' = -x + \mu \cdot (1 - x^2) \cdot y \end{cases} \quad (3)$$

Równanie van der Pola (1) zapisane w takiej postaci (3) można rozwiązać za pomocą metody Runge-Kutty, korzystając np. z arkusza kalkulacyjnego Excel. Poniżej przedstawiono algorytm rozwiązania układu równań, według 4 poziomowego schematu Runge-Kutty.

$$k_1 = y \quad (4)$$

$$m_1 = -x + \mu \cdot (1 - x^2) \cdot y \quad (5)$$

$$k_2 = y + h \cdot \frac{m_1}{2} \quad (6)$$

$$m_2 = -(x + h \cdot \frac{k_1}{2}) + \mu \cdot (1 - (x + h \cdot \frac{k_1}{2})^2) \cdot (y + h \cdot \frac{m_1}{2}) \quad (7)$$

$$k_3 = y + h \cdot \frac{m_2}{2} \quad (8)$$

$$m_3 = -(x + h \cdot \frac{k_2}{2}) + \mu \cdot (1 - (x + h \cdot \frac{k_2}{2})^2) \cdot (y + h \cdot \frac{m_2}{2}) \quad (9)$$

$$k_4 = y + h \cdot m_3 \quad (10)$$

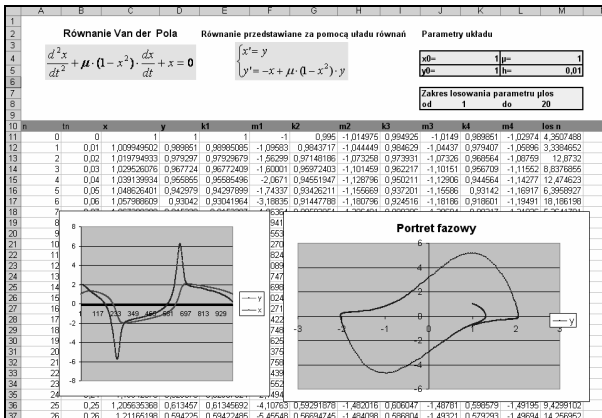
$$m_4 = -(x + h \cdot k_3) + \mu \cdot (1 - (x + h \cdot k_3)^2) \cdot (y + h \cdot m_3) \quad (11)$$

$$x_{n+1} = x_n + h \cdot (k_1 + 2 \cdot k_2 + 2 \cdot k_3 + k_4) / 6 \quad (12)$$

$$y_{n+1} = y_n + h \cdot (m_1 + 2 \cdot m_2 + 2 \cdot m_3 + m_4) / 6 \quad (13)$$

W metodzie tej musimy wielokrotnie obliczać wartości funkcji, korzystamy tutaj z poprzedniego kroku. Długość kroku całkowania określa h.

Poniżej przedstawiono widok arkusza kalkulacyjnego z widocznymi wynikami rozwiązania równania van der Pola z losowym parametrem  $\mu$ . Współczynnik tłumienia wynosi  $1 + \mu$  los.



Rys. 1 Widok arkusza kalkulacyjnego z widocznymi wynikami rozwiązania równania van der Pola z losowym parametrem  $\mu$ .

#### 4. PRZEPROWADZONE BADANIA OSCYLATORA VAN DER POLA

Głównym celem artykułu jest określenie stabilności układu opisanego za pomocą równania van der Pola (1). Przy czym wartość parametr  $\mu$  – czyli współczynnik tłumienia będzie liczbą losową z zakresu  $\langle 0;5 \rangle$ . Warunki początkowe natomiast będą stałe.

System jest strukturalnie stabilny, jeżeli przy niewielkiej zmianie parametrów zachodzą niewielkie zmiany w trajektoriach. Ze stabilnością strukturalną ściśle wiąże się pojęcie bifurkacji. Bifurkacją (łac. Bifurcus - rozdwojenie, rozwidlenie, rozdzielenie) nazywa się zjawisko, polegające na tym, że pod wpływem zmiany parametru następują jakościowe zmiany trajektorii. Punktem bifurkacji jest wartość parametru, dla którego układ traci stabilność.

Do określenia stabilności układu posłużymy nam portretu fazowy, w którym to będziemy szukali się przecinających się linii tego portretu. W momencie gdy linie portretu fazowego zaczną się przecinać układ przestaje być stabilny. Do przeprowadzenia badań wykorzystano aplikację napisaną za pomocą programu Borland Enterprise Suite C++ Builder. Wykorzystano metodykę programowania obiektowego. Stworzony program jest zdefiniowany za pomocą obiektów komunikujących się pomiędzy sobą w celu wykonywania zadań.

Wykorzystane komponenty:

TChart – jest jednym z najważniejszych komponentów udostępnionych przez bibliotekę TeeChart Zawiera on bogaty zbiór wykresów. Komponent ten służy do wizualizacji wyników.

TStringGrid – arkusz z elementami będącymi łańcuchami znaków. Komponent ten służy do wyświetlania wyników obliczeń w formie tabeli.

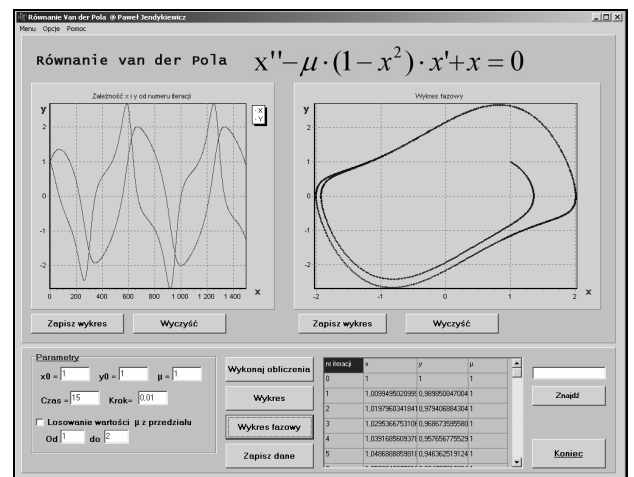
TCheckBox – pole wyboru, włączony i wyłączony. Za pomocą tego komponentu możemy wprowadzać losowość parametru  $\mu$  z przedziału.

MainMenu – menu związane z formularzem.. Komponent odpowiada za pasek menu w którym możemy zapisać dane, zapisać grafikę, zamknąć program.

Edit – edycja tekstu. Edit służy nam do wprowadzania parametrów.

Button – przycisk. Wykonywanie zdefiniowanych zadań.

W programie wykorzystano metodę Runge–Kutty czwartego rzędu opisaną w punkcie 3.



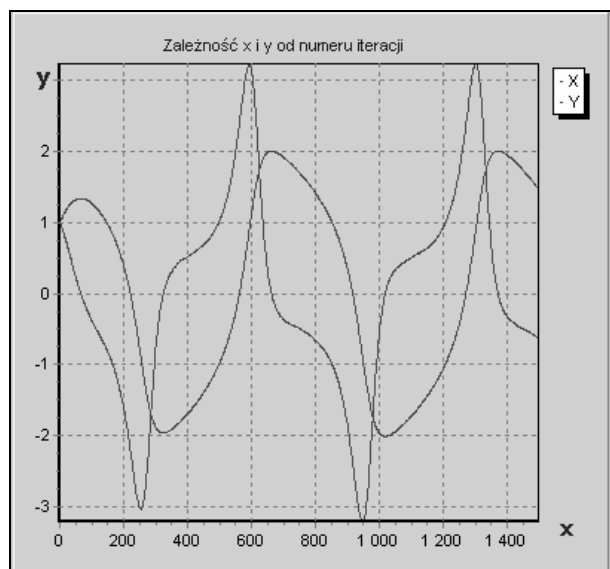
Rys. 2 Okno programu vdP

Program jest bardzo prosty w obsłudze. Po uruchomieniu programu możemy ustawiać parametry takie jak: wartości początkowe  $x_0, y_0$  wartość współczynnika tłumienia  $\mu$

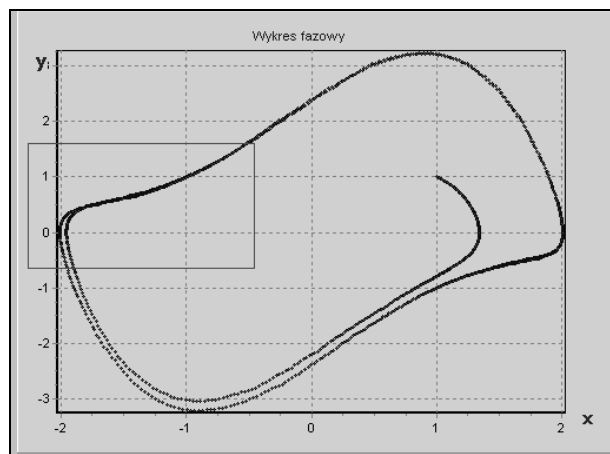
liczbę kroków  
czas symulacji.

**Symulacja nr.1**

Parametry układu: współczynnika tłumienia losowany z przedziału  $\langle 1;2 \rangle$ , warunki początkowe  $x_0=1$   $y_0= 1$ , czas symulacji =15, krok =0,01, ilość iteracji=1500.



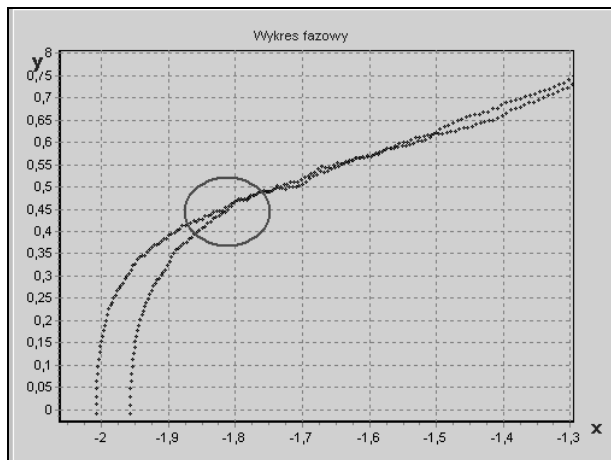
**Rys. 3** Otrzymany wykres składowych x i y oscylatora vdP.



**Rys. 4** Portret fazowy oscylatora vdP z parametrem losowym.

Widzimy ze portret fazowy równania (1) przy losowym współczynniku tłumienia z przedziału  $\langle 1;2 \rangle$  charakteryzuje się nierównościami trajektorii. Nierówności te spowodowane są zmienny współczynnikiem tłumienia. Skutkiem tego jest pojawienie się bifurkacji.

Aby zauważyć występujące bifurkacje wykorzystano opcje zoom, aby powiększyć interesujący fragment wykresu (rys.4).

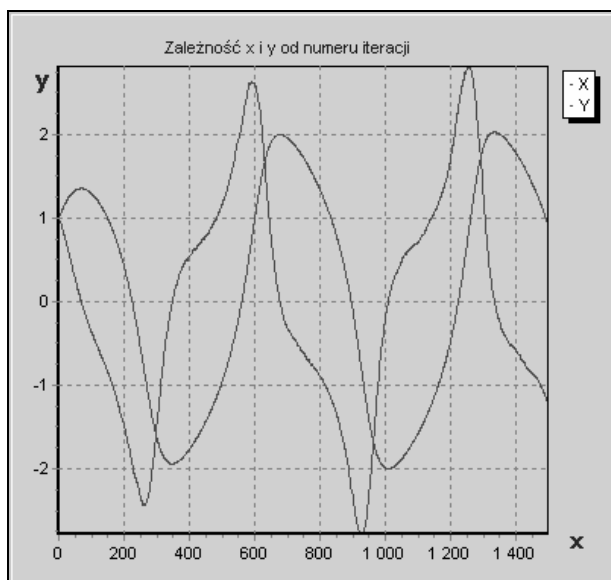


**Rys. 5** Powiększony fragment zaznaczony czerwoną ramką na rysunku 16.

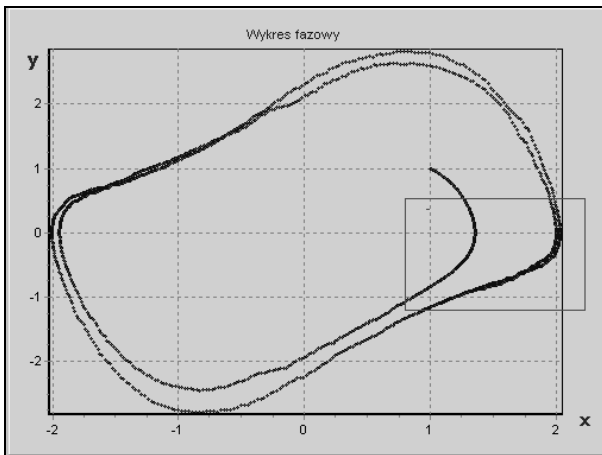
Na rysunku 5 portret fazowy przecina się, co oznacza że przekroczone została granica stabilności układu. Po przybliżeniu możemy zauważyć, że portret przecina się w punkcie  $(-1.819,0.447)$  co według tabeli odpowiada współczynnikowi tłumienia równemu  $\mu=1,0368$ .

**Symulacja nr. 2**

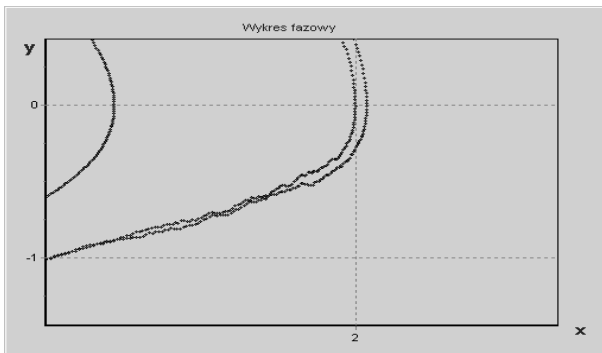
Parametry układu: współczynnika tłumienia losowany z przedziału  $\langle 0;2 \rangle$ , warunki początkowe  $x_0=1$   $y_0= 1$ , czas symulacji =15, krok =0,01, ilość iteracji=1500.



**Rys. 6** Otrzymany wykres składowych x i y oscylatora vdP



Rys. 7 Portret fazowy oscylatora vdp z parametrem losowym



Rys. 8 Powiększony fragment zaznaczony czerwoną ramką na rysunku 16.

Bifurkacje wystąpiły w punkcie  $(1.7634, -0.5974)$  co odpowiada współczynnikowi tłumienia  $\mu=1,86056$ .

## 5. PODSUMOWANIE

Najmniejsza losowa wartości współczynnika tłumienia przy której wystąpiły bifurkacje wynosi 0,49. Jest to granica po przekroczeniu której układ traci stabilność. Współczynnik tłumienia w tym układzie losowany był z przedziału  $<0;1>$ . Im większy przedział, z którego losowany jest współczynnika tłumienia tym jest większe prawdopodobieństwo wystąpienia bifurkacji. Na podstawie przeprowadzonych symulacji możemy stwierdzić, że losowy współczynnik tłumienia wpływa na stabilność układu. W ten sposób można przeprowadzać symulacje układów, w których elementy wraz z upływem czasu tracą swoje pierwotne właściwości. Co może powodować zaburzenia w układzie.

Praktyczne wykorzystanie metod numerycznych już od wielu lat wiąże się maszynami cyfrowymi. Współczesne metody numeryczne są rozwijane z myślą o ich realizacji na maszynach cyfrowych. Wykonanie

obliczeń numerycznych na komputerach tworzy nową klasę problemów. Wynikają one między innymi ze stosowanej arytmetyki. Każda liczba w pamięci komputera jest reprezentowana przez swoje skończone rozwinięcie. W każdym działaniu arytmetycznym powstaje błąd, ponieważ wynik jest również skończonym rozwinięciem. Dlatego nie należy przyjmować wyników bezkrytycznie i uznawać je za poprawne tylko dla tego, że zostały obliczone przez komputer.

## Literatura

1. Andrzej Polaczewski, Równania Różniczkowe zwyczajne. Wydawnictwo Naukowo Techniczne, Warszawa 1999.
2. Thompson J.M.T., Steward H. B. Nonlinear Dynamics and Chaos, Wiley & Sons 2002
3. van der Pol, B., van der Mark, J The heart beat considered as a relaxation oscillation, and an electrical model of the heart. Phil. Mag. 1928, 6, 763-775
4. Collatz L.: Metody numeryczne w rozwiązywaniu równań różniczkowych, Wydawnictwo Naukowe PWN, Warszawa 1982.
5. Jean Degras, Praktyczne metody analizy numerycznej. Wydawnictwo Naukowo Techniczne Warszawa 1974.
6. Kapitaniak T., Wojewoda J., Bifurkacje i chaos, Wydawnictwo Politechniki Łódzkiej, Łódź 1994.
7. E. M. Elabbasy, Synchronization of van der Pol oscillator and Chen chaotic dynamical system.
8. Awrejcewicz Jan, Chaos i synchronizacja w układach fizycznych, Wydawnictwo Politechniki Łódzkiej, Łódź 1997.
9. Awrejcewicz Jan, Tajemnice nieliniowej dynamiki, Wydawnictwo Politechniki Łódzkiej, Łódź 1997.

## INTERNET

10. [http://en.wikipedia.org/wiki/Van\\_der\\_Pol\\_oscillator](http://en.wikipedia.org/wiki/Van_der_Pol_oscillator)
11. [http://www.its.caltech.edu/~mcc/Chaos\\_Course/Lesson3/Demos.html](http://www.its.caltech.edu/~mcc/Chaos_Course/Lesson3/Demos.html)
12. <http://mathworld.wolfram.com/vanderPolEquation.html>
13. <http://cyfbar.republika.pl/spistresci.htm>



# OBLICZENIA LUDZKIE JAKO SPOSÓB NA ROZSZERZANIE BAZ WIEDZY NA PODSTAWIE GRY 'ZGADNIJ AUTORA'

**Tomasz Smykowski**

Uniwersytet Kazimierza Wielkiego  
Instytut Mechaniki Środowiska i Informatyki Stosowanej  
III rok Informatyki  
ul. Chodkiewicza 30, 85-064 Bydgoszcz  
email: tomasz\_smykowski@o2.pl

**Streszczenie:** *Praca ta przedstawia wyniki gry online 'Zgadnij autora' jako przykładu gry opierającej się na zasadach human - based computation. Gra ta polega na odgadywaniu autora książki. Dzięki prostym, jasnym zasadom i dodatkowym elementom wpływającym na atrakcyjność dla graczy (nagrody, rywalizacja itp.) udało się osiągnąć znaczące efekty w gromadzeniu bazy danych połączeń autor – książka. Są to dane dostępne tradycyjnymi sposobami, ale ciężko osiągalne. Oprócz tego udało się wypracować mechanizmy i sprecyzować założenia jakie gry human - based computation muszą spełniać, aby były atrakcyjne dla użytkowników oraz jak zapobiegać dodawaniu błędnych danych do bazy.*

**Słowa kluczowe:** : human - based computation, obliczenia ludzkie, gra, zgadnij autora, baza wiedzy, autor, książka

## Human - based computation as a way to broaden knowledge bases. The case of the 'Guess the author' game

**Abstract:** *Abstract: This article presents the results of an on-line game called "Guess the author" as an example of a game grounded on human - based computation. The game consists in guessing authors of books. Thanks to its easy and clear rules as well as additional features that make the game more attractive to players (like awards, competition etc.), the game has been successful in collecting a data base of author – book links. This data is available in conventional way but rather hard to achieve. Apart from that, the authors successfully developed mechanisms and specified conditions that have to be met by human - based computation games so as to make them attractive and to prevent adding incorrect data to the database.*

**Keywords:** human - based computation, game guess the author, knowledge base, author, book

### 1. WPROWADZENIE

Technika *human – based computation* polega na przekazywaniu ludziom części obliczeń do wykonania przez komputery, które same nie mogą ich zrealizować. Dzięki połączeniu celów naukowych i atrakcyjności narzędzi można połączyć dużą liczbę osób w celu powiększania bazy danych o informacje, które w żaden inny sposób nie mogą zostać zgromadzone. Dane takie nie mogą zostać wprowadzone przez narzędzia automatyzacji lub przez inne algorytmy komputerowe w sposób efektywny. [1]

Gra 'Zgadnij autora' została umieszczona na stronie internetowej założonej przez autora. Została ona napisana w języku *PHP* z użyciem bazy danych *MySQL*.

Gra polega na odgadywaniu autorów książek podawanych przez innych użytkowników.

Celem podstawowym gry jest analiza możliwości human – based computation w zastosowaniu w następnych przedsięwzięciach związanych z gromadzeniem wiedzy, m.in. na temat hipernimii i znaczenia wyrazów. Powiązania autor – książka są możliwe do uzyskania przez narzędzia automatyzacji, ale jest to utrudnione. Dlatego autor postanowił użyć takiego celu do sprawdzenia możliwości metody human – based computation.

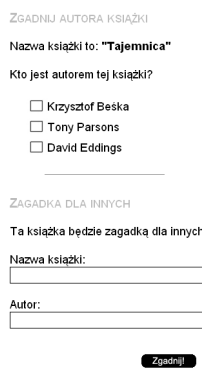
### 2. ZASADY GRY

Ilustracja 1 przedstawia formularz gry. Składa się on z 4 części, przy czym część 'Zgadnij autora książki' i 'Zgadka dla innych' dotyczą samej rozgrywki. Po lewej

stronie w części 'Wynik' znajdują się informacje statystyczne oraz 'Zasady gry' ze zbiorem reguł gry dla początkujących graczy. Jak widać na załączonych ilustracjach oprawa graficzna gry jest czytelna i atrakcyjna wizualnie dla odbiorcy.



Rys. 1 Widok na cały ekran gry 'Zgadnij autora'.



Rys. 2 Przykładowy formularz rozgrywki, który mogą zobaczyć uczestnicy gry.

Zasady gry są następujące:

Autorów trzeba podawać bez skracania imion. Jeżeli nie ma polskiego odpowiednika, trzeba podać nazwę angielską. Za każdego odgadniętego autora otrzymuje się 1 punkt. Za nie odgadniętego autora nie traci się punktów. Jeżeli w zagadce jest błąd, trzeba nacisnąć "Zgadnij", aby przejść do następnego pytania. W przypadku nieznamości odpowiedzi należy odświeżyć stronę, aby przejść do kolejnego pytania. Brak odpowiedzi na pytanie zagadki nie powoduje utraty punktów. Grę można kontynuować w każdej chwili, nawet po ponownym zalogowaniu się na stronę. W celu zapewnienia jakości gry, konta osób naruszających ciężko powyższe zasady są blokowane.

Zasady te zostały wypracowane w trakcie udziału w grze uczestników. Są one kompromisem między wymaganiami systemu gromadzenia danych, możliwościami ich automatycznej obróbki i atrakcyjności gry dla uczestników, co zostanie objaśnione szczegółowo w rozdziale 4. Warto tutaj zauważyć, że przejrzystość zasad jest istotna aby uniknąć kontrowersji wobec ich sprawiedliwości. [2], [3]

Użytkownik ma dostęp do statystyk dostarczających informacji o tym, na którym miejscu znajduje się w rankingu gry oraz ile uzyskał punktów. Na stronie rankingu znajduje się informacja o 10 najlepszych użytkownikach od początku trwania gry i o 10 użytkownikach najlepszych bieżącego dnia:

NAJLEPSI DZISIAJ		
Miejsce	Użytkownik	Poprawnych odpowiedzi
1	pearl	5

NAJLEPSI Z NAJLEPSZYCH		
Miejsce	Użytkownik	Poprawnych odpowiedzi
1	marcysia	1055
2	bejs2	892
3	Effa	549
4	kiniunia25	504
5	lakie	405
6	peka	274
7	Ela	270
8	ghesam	248
9	ala	245
10	renataszagabika	176

Rys. 3 Ranking gry.

### 3. SZCZEGÓLOWE CELE GRY

Cele gry dzielą się na dwa rodzaje. Pierwszy z nich dotyczy tego, o jakie dane ma zostać wzbogacona baza danych przez udział uczestników. Drugi rodzaj to cele, które gra realizuje dla samych uczestników i wpływa na ich satysfakcję z udziału w niej. Celem gry z punktu widzenia autora jest zgromadzenie bazy książek, czyli połączeń autor – książka. Największa strona o książkach w Polsce posiada przeszło 30 tys. autorów i 90 tys. książek. Autor postawił za cel do realizacji otrzymanie 10 tys. wpisów samoweryfikujących się. Zakładając, że 80% wpisów będzie weryfikować 20% pozostałych. Założono zatem osiągnięcie 2 tys. poprawnych i zweryfikowanych powiązań autor – książka, co spełniłoby *zasadę Pareto*. [4]

Celem gry dla użytkowników jest chęć weryfikacji swojej wiedzy na temat książek. Sprawdzenie, jak dużo książek się pamięta. Cel ten jest realizowany przez udział w grze i otrzymane statystyki. Celem jest realizacja i porównanie swojej wiedzy z innymi uczestnikami gry – do osiągnięcia tego służy ranking 10 najlepszych uczestników. W trakcie gry dodano także ranking 10

najlepszych uczestników bieżącego dnia w celu pozytywnego wzmocnienia motywacji uczestników, którzy na początku muszą zdobyć relatywnie dużo punktów, aby trafić do głównego rankingu.

Celem gry jest czysta zabawa – udział w grze nie wymaga żadnych specjalnych umiejętności poza wiedzą o powiązaniach książka – autor. Oprócz tego w trakcie gry podjęto próby znalezienia sponsorów nagród w celu zwiększenia atrakcyjności gry dla użytkowników. Uzyskano wsparcie w postaci nagród rzeczowych.

Aby uzyskać odpowiednio dużą bazę samo weryfikujących się danych gra musi spełniać jak najlepiej cele stawiane przez użytkowników [5]. W ten sposób gra jest dla nich atrakcyjna i chętnie biorą w niej udział. W trakcie tworzenia gry napotkano na problem związany ze zwiększeniem popularności pomysłu i samej gry, co wymusiło usprawnienie szybkości działania procesów weryfikacji danych. Jednak liczba 239 zarejestrowanych użytkowników w kontekście tych problemów jest odczytywana jako sukces. Pod uwagę trzeba wziąć też, na czym polega gra – mimo skierowania jej do wąskiej grupy osób, osiągnięto dosyć dużą bazę danych. Baza ta zostanie opisana w dalszej części tego opracowania.

#### 4. PROBLEMY I ROZWIĄZANIA

W czasie gry rozpoznano kilka problemów związanych z wprowadzaniem danych przez użytkowników. Pierwotnie gra polegała na podawaniu autora w polu tekstowym. Z tego powodu, aby dostać punkt trzeba było podać dokładnie takie same nazwisko jak autor zagadki. Z powodu błędów dokonywanych przez użytkowników ten sposób nie okazał się skuteczny. Zostało to zastąpione wyborem z listy trzech autorów z powodu możliwych błędów w pisowni autorów.

Oprócz tego pojawiły się problemy ze skracaniem przez użytkowników nazw autorów albo nieznanomość ich pełnych nazwisk. Problem ten został rozwiązany przez rozszerzenie regulaminu i dodanie blokady dodawania autorów ze skróconymi nazwiskami.

Następny problem to wprowadzanie danych w złe pola: w pole autora – nazwy książki, a w pole nazwy książki – autora. Pojawiły się także niedozwolone znaki – np. nawiasy klamrowe, cudzysłowie itp. Dodatkowo pojawiały się wpisy bez polskich liter, drobne literówki, nieużywanie dużych liter w nazwach książek i nazwiskach autorów. Te problemy zostaną rozwiązane na etapie automatycznej weryfikacji.

Oprócz powyższych pojawili się użytkownicy, którzy dodający fałszywe dane, tylko w celu zanieczyszczenia bazy danych i uzyskania wysokich miejsc w rankingu. Aby zabiec temu zjawisku przewidziano następujące

środki zaradcze: aby wziąć udział w grze trzeba się zarejestrować, natomiast aby dodać nową książkę, trzeba odgadnąć autora innej książki.

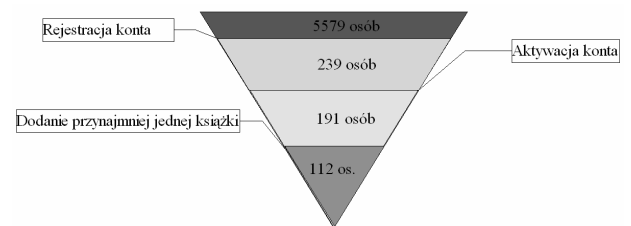
Drugi mechanizm jest związany też z samo weryfikacją danych, naturalną konsekwencją tego, że gra sama w sobie nie jest łatwa, jest to, że użytkownicy wprowadzający fałszywe dane szybko sami rezygnują z tych działań. Przygotowane narzędzia automatyczne i pół-automatyczne powodują wychwytywanie takich użytkowników i blokowanie im kont, albo usuwanie błędnych wpisów, jeżeli są one sporadyczne.

Samo weryfikacja wpisów polega na tym, że można na podstawie zebranych danych automatycznie wybrać poprawne wpisy. Elementami samo weryfikacji są:

- ile osób niezależnie wprowadziło takiego samego autora i nazwę książki
- ile osób odgadło / nie odgadło poprawnie autora podanej książki
- ile poprawnych książek wprowadził użytkownik do tej pory

#### 5. WYNIKI PRZEDSIĘWZIĘCIA

Przedsięwzięcie trwało od 25 grudnia 2007 i trwa nadal. Powyższe opracowanie jest przygotowane na podstawie danych z dnia 1 maja 2008, czyli z 5 miesięcznego okresu czasu. Stronę odwiedziło 5579 internautów, zarejestrowało się 239 użytkowników. Spośród tych osób 48 nie aktywowało swojego konta przez kliknięcie w link aktywacyjny w wiadomości email, którą otrzymali.



Rys. 4 Odwrócona piramida liczby osób, które podjęły działania związane z grą.

Z sumy 239 użytkowników z aktywowanymi kontami w grze uczestniczyło  $239 - 127 = 112$  osób. Oznacza to, że tylko 46% osób, które założyły konto na stronie aktywowało swoje konto i dodało przynajmniej jedną książkę. Średnia ta sprawdzała się przez większość czasu trwania gry.

Co druga osoba zrezygnowała z udziału, Wykluczone zostały więc osoby, które nie są przekonane i zdecydowane brać udziału w grze, której zasady zostały ściśle określone. Z punktu widzenia celów naukowych

uczestnictwo tych osób mogłoby doprowadzić do zebrania w bazie błędnych lub niespójnych danych.

W ciągu 5 miesięcy użytkownicy uczestniczący w grze dodali 8369 zagadek. Pierwsza osoba z rankingu wprowadziła 1055 zagadek, druga 892, trzecia 549, czwarta 504, piąta 405, szósta 274, siódma 270, ósma 248, dziewiąta 245, a dziesiąta 176.

Średnia wprowadzonych zagadek to około 60 na aktywnego uczestnika gry. Średnio do gry dołącza 22 aktywnych uczestników miesięcznie. Każda z tych zagadek to odpowiedź na zagadkę i wprowadzenie nowej zagadki. Ponieważ dane te powielają się, rzeczywista liczba wprowadzonych książek jest mniejsza od tej liczby.

Analiza danych wskazała jednak, że z 8369 wpisów można wyekstrahować 82% poprawnych połączeń autor-książka tj. 7380. Oznacza to, że 82% wpisów jest poprawnych i potwierdzonych przez pozostałe wpisy. W konsekwencji w ciągu 5 miesięcy użytkownicy uzupełnili bazę o 7380 książek.

## 6. PODSUMOWANIE

Na wstępie założono zebranie 2000 poprawnych wpisów autor - książka. Oznaczałoby to realizację zasady Pareto. Okazało się, że zasada ta została zrealizowana, ale w wariacie optymistycznym. Użytkownicy wprowadzili przeszło 8369 wpisów, z czego poprawne to 7380. Wynika stąd, że 18% weryfikuje pozostałe 82% wpisów. Ponieważ badanie to było pilotażowe, nie przewidywano dużej promocji gry 'Zgadnij autora'. Mimo to uzyskane dane eksperymentalne wskazują na pewne prawidłowości w zastosowaniu podejścia human – based computation w gromadzeniu danych. Jest to metoda efektywna przy dobrym przygotowaniu gry.

Mimo najlepszego algorytmu gry tego typu nie będą przydatne, jeżeli nie będą efektywne dla użytkowników. Ma to tym bardziej duże znaczenie, ponieważ aby uzyskać bazę wiedzy na tematy docelowe: czyli języka, lingwistyki, hipernimii itp. będzie potrzebna dużo większa grupa osób.

Cel ten można osiągnąć przez znaczny nacisk na jakość gry i reklamę. Dzięki dużej reklamie, współpracy ze sponsorami nagród i mediami jest szansa na przyciągnięcie do gry human - based computation dużej liczby odbiorców i chętnych do udziału w niej.

Daje to według autora ogromne perspektywy w zgromadzeniu bazy wiedzy w dziedzinach, w których prace badawcze napotykają na problemy wynikające z:

- - dużej liczby informacji do wprowadzenia
- - ograniczonych możliwości wsparcia procesu wprowadzania danych przez komputery

- - wymaganej dużej liczby uczestników projektu przy stosunkowo niskich możliwościach finansowych
- - potrzeby weryfikacji danych wprowadzanych przez uczestników projektu.

## Literatura

1. Luis von Ahn, *Human Computation*, ACM, New York, USA (2007)
2. Luis von Ahn and Laura Dabbish, *Labeling Images with a Computer Game*, ACM, New York, USA (2004)
3. Zach Whalen, Google Image Labeler – Game Theory and Information, [http://www.gameology.org/blog/google\\_image\\_labeler\\_game\\_theory\\_and\\_information](http://www.gameology.org/blog/google_image_labeler_game_theory_and_information),
4. Steve Prentice, *Cool Time: A hands-on Plan for Managing Work and Balancing Time*, John Wiley and Sons, San Francisco (2006)
5. L. von Ahn, S. Ginosar, M. Kedia, R. Liu and M. Blum, *Improving Accessibility of the Web with a Computer Game*, ACM, New York, USA (2006)

# IDEA SKIEROWANYCH LICZB ROZMYTYCH - OBSERWACJA ROZMYTA

**Dorota Wilczyńska-Sztyma**

Uniwersytet Kazimierza Wielkiego  
Instytut Mechaniki Środowiska i Informatyki Stosowanej  
doktorantka  
ul. Chodkiewicza 30, 85-064 Bydgoszcz  
email: dorotaws@ukw.edu.pl

**Streszczenie:** Niniejsze opracowanie ma stanowić przyczynek do szerszej refleksji nad sensem powstania skierowanych liczb rozmytych (ang. Ordered Fuzzy Numbers). Przykłady te mają przekonać nie tylko do nowatorskiego podejścia do algebry rozmytej, ale także ukazać sens istnienia skierowania.

**Słowa kluczowe:** : Skierowane liczby rozmyte, obserwacja rozmyta, logika rozmyta

## The idea of ordered fuzzy numbers – fuzzy observation

**Abstract:** This paper is intended to represent a valuable contribution to broader reflection on the sense of creating Ordered Fuzzy Numbers. Presented examples are aimed at making others warm to the innovative attitude towards fuzzy algebra but also at showing the sense of ordering itself.

**Keywords:** Ordered fuzzy numbers, fuzzy observation, fuzzy logic

### 1. WSTĘP

Człowiek często stosuje pojęcia nie ostre i nie precyzyjne jak "wysoka jakość materiału", „niski wzrost gospodarczy”, „średnie zasolenie” itp. Choć nie zawierają w sobie danych liczbowych dla innego człowieka są rozumiane jednoznacznie. Koncepcja zbiorów rozmytych jest jedną z dziedzin sztucznej inteligencji. Teoria ta w szczególności sposób przydatna jest w przypadku systemów, w których czynnik ludzki odgrywa zasadniczą rolę. Należą do nich ekonomia, medycyna, socjologia, teoria podejmowania decyzji, ale także dziedziny techniki. Jest pewnego rodzaju tłumaczem języka nieprecyzyjnych i nie ostrych pojęć na język liczb.

### 2. SKIEROWANE LICZBY ROZMYTE

Model skierowanych liczb rozmytych został zaproponowany przez prof. W. Kosińskiego, dr. P.

Propokowicza oraz dr. D. Ślęzaka w 2002 roku. Założeniem ich było wyeliminowanie wad klasycznej algebry liczb rozmytych opartej na zasadzie rozszerzania według prof. L. Zadeha. Wady te to przede wszystkim: zwiększanie rozmytości przy wykonywaniu kolejnych działań, brak możliwości wnioskowania wstecz oraz duże skomplikowanie obliczeniowe. W dalszej części artykułu będzie używany skrót OFN (ang. Ordered Fuzzy Numbers) na określenie skierowanych liczb rozmytych. By móc prowadzić rozważania na temat idei skierowanych liczb rozmytych należy przybliżyć ich definicję.

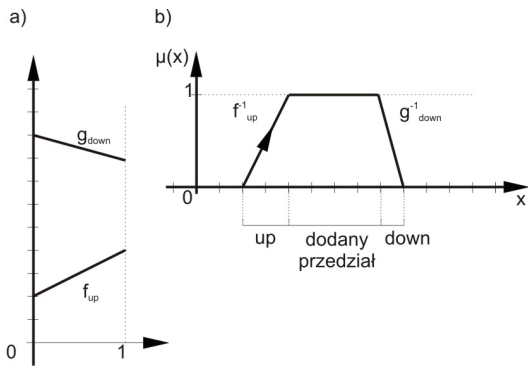
**Definicja 1** Skierowaną liczbą rozmytą  $A$  nazywamy uporządkowaną parę  $(f, g)$  funkcji takich że

$f, g : [0, 1] \rightarrow \mathbb{R}$  są ciągłe.

Funkcje te nazywają się odpowiednio: częścią (ramieniem) up i częścią down. Funkcje  $f$  i  $g$  są połączone przedziałem  $(\text{const})$  zawierającym się pomiędzy  $[f(1), g(1)]$  i równym 1. Granice funkcji oznaczają się  $\text{up} = (1A, 1-A)$  oraz  $\text{down} = (1+A, pA)$ .

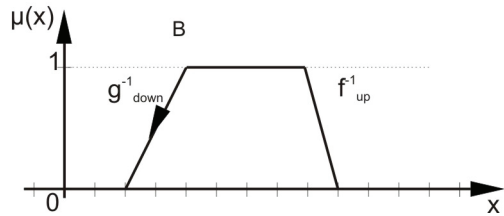
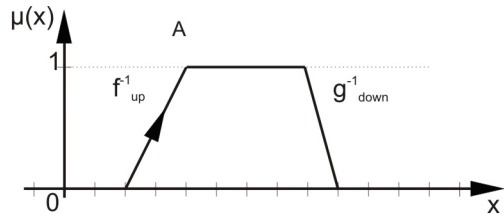
Granice te są liczbami rzeczywistymi. Nie muszą one być właściwe. Znaczy to tyle, że nie muszą spełniać warunku  $1_A \leq 1 - A$  oraz  $1 + A \leq p_A$  (rys. 3). Przyporządkowuje się kolejno:

$$l_A := f(0), l^-_A := f(1), l^+_A := g(1), p_A := g(0)$$



**Rys. 1 a)** Interpretacja graficzna OFN, **b)** OFN przedstawiona w sposób nawiązujący do klasycznych liczb rozmytych z dodatkową strzałką ukazującą skierowanie (orientację) liczby. Skierowaniem liczby rozmytej nazywa się uporządkowanie funkcji f i g tak, że funkcja f jest początkiem OFN, zaś g końcem tej liczby (porównaj rys. 2). W zależności od skierowania OFN można podzielić na dwa typy:

- o skierowane liczby rozmyte o orientacji pozytywnej wtedy, gdy kierunek OFN jest zgodny z kierunkiem osi X liczb rzeczywistych. Dla właściwej OFN o orientacji pozytywnej przyporządkowuje się kolejno:  
 $l_A := f(0), l^-_A := f(1), l^+_A := g(1), p_A := g(0)$
- o skierowane liczby rozmyte o orientacji negatywnej wtedy, gdy kierunek OFN jest przeciwny do kierunku osi X liczb rzeczywistych. Dla właściwej OFN o orientacji negatywnej przyporządkowuje się kolejno:  
 $l_A := g(0), l^-_A := g(1), l^+_A := f(1), p_A := f(0)$



**Rys. 2** OFN A i B z tym samym kształtem funkcji, lecz o przeciwnym skierowaniu.

Żeby w odpowiedni sposób przedstawić innowacje, należy przyrzeć się możliwości jakiego algebra OFN. Zasady podstawowych działań zostaną przybliżone niżej. Niech  $A=(f_A, g_A)$ ,  $B=(f_B, g_B)$ ,  $C=(f_C, g_C)$  będą skierowanymi liczbami rozmytymi wtedy:

- o suma  $C=A+B$   
 $f_C = f_A + f_B, g_C = g_A + g_B$
- o różnica  $C=A-B$   
 $f_C = f_A - f_B, g_C = g_A - g_B$
- o iloczyn  $C=A*B$   
 $f_C = f_A * f_B, g_C = g_A * g_B$
- o iloczyn przez skalar  $C=A*r$   
 $f_C = r*f_A, g_C = r*g_A$
- o iloraz  $C=A/B$   
 $f_C = f_A / f_B, g_C = g_A / g_B$

W przeciwieństwie do klasycznych liczb rozmytych rezultatem operacji na OFN nie zawsze jest liczba o większym nośniku. Wykonując operację arytmetyczną można otrzymać liczbę, której nośnik jest nieskończenie mały, interpretuje się ją jako liczbę rzeczywistą. Taka liczba nazywa się singletonem, tak jak w klasycznych liczbach rozmytych i traktuje się ją jako wyjątkowy przypadek liczby skierowanej rozmytej.

$$A-A=0 \text{ oraz } A*1/A=1$$

Tak jak w arytmetyce liczb rzeczywistych, analogiczne operacje na OFN są przemienne jak i łączne oraz mnożenie jest rozłączne względem dodawania. Co nie było takie oczywiste dla algebry klasycznych liczb rozmytych opartej na zasadzie rozszerzania. Dzięki temu jest się w stanie rozwiązać równania, które nie miało rozwiązania dla klasycznych liczb rozmytych.

$$X=C-A$$

Używanie liczb OFN w przeciwieństwie do klasycznych liczb rozmytych nie zwiększa nośnika kolejnych wyników liczb. Dzięki temu mogą one przechodzić

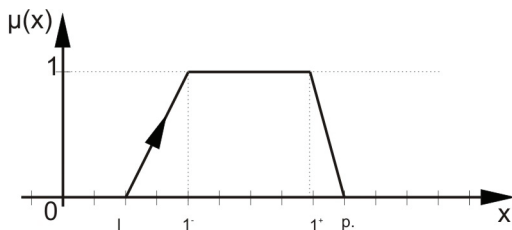
przez wiele operacji nie tracąc przy tym na dokładności. Dają one przy tym możliwość wnioskowania wstecz. Dodatkowo idea skierowanych liczb rozmytych łączy w sobie operowanie na liczbach rozmytych jak i rzeczywistych.

### 3. OBSERWACJA ROZMYTA

Codziennie jest się obserwatorem procesów zachodzących w otoczeniu: natężenie ruchu samochodów w mieście, rotacja towaru w sklepie. Każdy z tych procesów jest trwający w czasie. Patrząc z perspektywy czasu, wschód i zachód słońca w stosunku do natężenia promieniowania słonecznego są sobie przeciwnie. W kolejnych jednostkach czasu obserwuje się zdarzenia, którym przypisuje się oceny w skali [0,1]. Obserwację taką i zmiany w czasie można nazwać obserwacją rozmytą. Czas lub też monotoniczność (narastanie, zanikanie) zjawiska może być dla projektanta systemu decydującym czynnikiem wpływającym na skierowanie liczb w jego projekcie. W obserwacji rozmytej istnienie czasu może być wykorzystane jako możliwość wyboru parametru dla przedstawienia krzywej przynależności skierowanej liczby rozmytej w postaci parametrycznej. Jeśli para funkcji ciągłych  $(f, g)$  reprezentuje skierowaną liczbę rozmytą  $A$ , to przy wyborze  $t$  jako parametru czasu obserwacji rozmytej, której wynikiem jest  $A$ , można byłoby przedstawić krzywą przynależności odpowiadającej liczbie  $A$  w postaci pary funkcji:

$$\begin{aligned} x &= \hat{x}(t), y = \hat{y}(t), t \in [t_0, t_f] \\ t_1 &\leq t_{1+} \\ \hat{y}(t_1) &= \hat{y}(t_{1+}) = \hat{y}(t) = 1, t \in [t_1, t_{1+}] \\ f(\hat{y}(t)) &= \hat{x}(t) = \hat{y}(t), \text{ gdy } t \in [t_0, t_1] \\ g(y(t)) &= x(t), \text{ gdy } t \in [t_1, t_f] \end{aligned}$$

Przedstawione zostanie kilka przykładów, które dowodzą sensu zaistnienia OFN nie tylko jako obiektów matematycznych, ale również jako pewnych prognoz, tendencji, ocen, parametrów. W przykładach posłużono się modelem liczb trapezoidalnych i trójkątnych - funkcje  $f$  i  $g$  są funkcjami liniowymi (rys. 1, 2, 3).

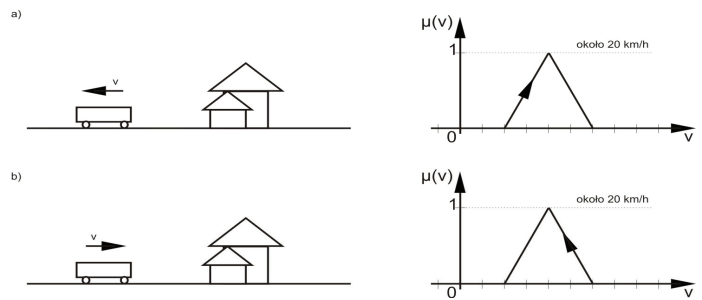


Rys. 3 Trapezoidalna OFN.

Każdą trapezoidalną OFN można opisać za pomocą charakterystycznych punktów (granice te opisano powyżej):  $l, l^-, l^+, p$  (rys. 3), gdzie punkt  $l$  zawsze znajduje się z lewej strony danej liczby niezależnie od jej skierowania. By jednak uwidocznic skierowanie używany będzie zapis dla liczby o orientacji dodatniej  $(f(l), f(l'), g(l'), g(0))$  oraz dla OFN o orientacji ujemnej  $(g(0), g(l), f(l), f(0))$ . W podobny sposób są traktowane trójkątne OFN, z tym, że punkty  $l^-$  oraz  $l^+$  są sobie równe. W ich przypadku zapis zostanie ograniczony do następujących trzech liczb  $(f(0), f(l), g(0))$ .

#### Przykład 1

Rozważany będzie przypadek, w którym obserwator znajdujący się na pewnym dworcu szacuje, że pociąg porusza się z prędkością  $v \approx 20 \text{ km/h}$ . Jednak to nie jest wystarczająca informacja. Dla obserwatora ważne jest również czy pociąg będzie się zbliżał, czy oddalał od dworca (rys. 4), czy uciekł mu, czy dopiero nadjeżdża.



Rys. 4 Pociąg a) oddalający się

b) zbliżający się od dworca z prędkością około 20 km/h

Chcąc zapisać prędkość jako klasyczną liczbę rozmytą nie ma możliwości zaznaczenia w jakim kierunku postępuje ruch. OFN dają tę możliwość w postaci skierowania. Skierowanie jest tu prognozą co się wydarzy w następnej chwili czasu czy odległość od dworca będzie rosła czy malała. Dzięki OFN, obserwator może nie tylko oceniać, w jakim stopniu uznaje prawdziwość rozpatrywanego zjawiska, ale także wyrazić swoją ocenę dotyczącą jego dynamiki.

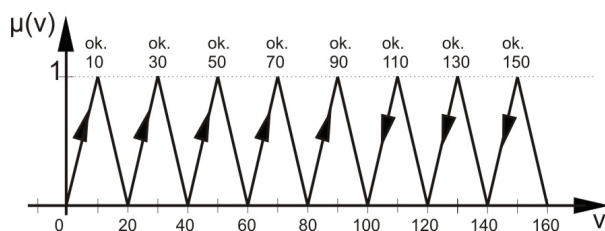
#### Przykład 2

Rozważony zostanie uproszczony model działania tempomatu. Jest to urządzenie pozwalające na utrzymywanie określonej prędkości, bez znaczenia czy

pojazd porusza się po płaskiej czy też pochylej nawierzchni. Dzięki niemu prowadzący pojazd nie musi naciskać pedału przyspieszenia, ponieważ odbywa się to automatycznie. Rozważone zostanie sens użycia skierowanych liczb do sterowania, które się odbywa w tym urządzeniu, co za tym idzie przyspieszeniem bądź hamowaniem pojazdu. Oczywiście rozważany przykład jest teoretyczny.

Rozważone będzie osiem OFN obranych dla kontinuum prędkości w przypadku, w którym prędkość jest mierzona z dokładnością do 0,2 km/h. Będą one wyglądały następująco: *około 10* (0, 10, (20,1)), *około 30* (20, 30, (40,1)), *około 50* (40, 50, (60,1)), *około 70* (60, 70, (80,1)), *około 90* (80, 90, (100,1)), *około 110* (100, 110, (120,1)), *około 130* (120, 130, 140). Wszystkie OFN będą na początku tego samego dodatniego skierowania. Zbiory liczb się pokrywają w pewnym zakresie, by uniknąć sytuacji, w której wystąpi wartość prędkości z pomiędzy nich, dla której minimalna wartość funkcji przynależności do każdego z nich była by równa zero. Zakres tych nałożeń jest mniejszy niż dokładność odczytu co daje w rezultacie to, iż każda wartość odczytywanej prędkości będzie należała tylko do jednej OFN z przedziału (0, 140).

W pewnym momencie prowadzący pojazd wybrał prędkość z jaką chciałby się przemieszczać na 100 km/h. Skierowanie wszystkich OFN położonych powyżej tej wartości prędkości zmieniło kierunek na przeciwny do pozostałych (rys. 5). O tej chwili przedstawiają się następująco: *około 110* ((120,1), 110, 100), *około 130* (140, 130, 120).



Rys. 5 OFN założone dla prędkości pojazdu.

Baza reguł oparta na wnioskowaniu typu Uogólniony Modus Ponens przedstawia się następująco:

1. jeśli  $v$  jest *około 10* (0, 10, 20,1) to  $a$  jest  $5m/s^2$ ;
2. jeśli  $v$  jest *około 30* (20, 30, 40,1) to  $a$  jest  $4m/s^2$ ;
3. jeśli  $v$  jest *około 50* (40, 50, 60,1) to  $a$  jest  $3m/s^2$ ;
4. jeśli  $v$  jest *około 70* (60, 70, 80,1) to  $a$  jest  $2m/s^2$ ;
5. jeśli  $v$  jest *około 90* (80, 90, 100,1) to  $a$  jest  $1m/s^2$ ;
6. jeśli  $v$  jest *około 110* (120,1, 130, 100) to  $a$  jest  $-1m/s^2$ ;
7. jeśli  $v$  jest *około 130* (140, 130, 120) to  $a$  jest  $-2m/s^2$ .

Jeśli wartość prędkości pojazdu będzie się zawierać w OFN o skierowaniu dodatnim to przyspieszenie  $a$  będzie przybierało wartości dodatnie zaś jeśli wartość prędkości będzie się zawierać w OFN o skierowaniu dodatnim to przyspieszenie  $a$  będzie przybierało wartości ujemne. Wartość przyspieszenia liczb *około 90* i *około 110* oraz liczb *około 70* i *około 130* będzie przybierała tą samą wartość, ale inny znak. Skierowania interpretowane jest w tym przypadku jako wyznacznik znaku wartości funkcji pierwotnej po czasie. Informuje ono z góry jaki znak będzie przybierało przyspieszenie  $a$ , krótko mówiąc w przypadku samochodu znaczyło by to tyle, który pedał automat ma docisnąć: gazu czy hamulca.

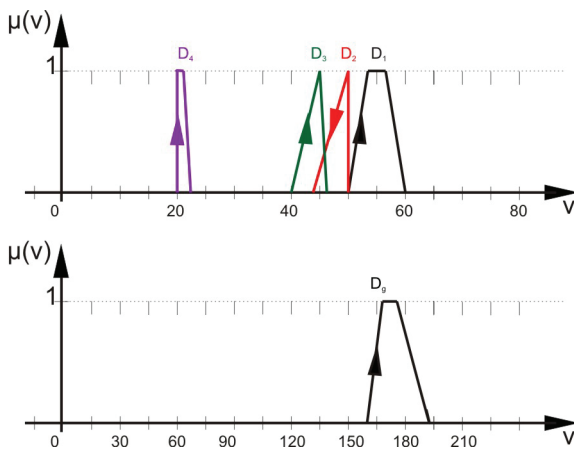
### Przykład 3

Dotyczy dziedziny, z którą każdy ma coś wspólnego czyli finansów. Próby zastosowania OFN w ekonomii można znaleźć w publikacji [3], gdzie przeanalizowano model Leontiewa przy użyciu OFN dla opisu współrzędnych wektora produkcji całkowitej. Przybliżony tu model będzie prostszy w interpretacji.

Tabela 1 Zestawienie kosztów i przychodów

M(i)	Przewidywany przychód ( $P_i$ )	Przewidywane koszty ( $K_i$ )	Przewidywany dochód ( $D_i = P_i - K_i$ )
1	(100, 108, 116, 130)	(50, 55, 60, 70)	(50, 53, 56, 60)
2	(120, 130, 140, 145)	(70, 80, 90, 101)	(50, 50, 50, 44)
3	(80, 75, 70, 60)	(40, 30, 25, 22)	(40, 45, 45, 46)
4	(105, 100, 96, 92)	(85, 80, 75, 70)	(20, 20, 21, 22)
Obliczony dochód globalny			(160, 168, 172, 194)

Teoretyczna firma posiada sieć czterech marketów. Zostały one ponumerowane  $i=1, 2, 3, 4$ . Na sprzedaż globalną  $D_g$  firmy składa się przychód ze sprzedaży produktów  $P_g$  minus koszty  $K_g$  tj.: cena zakupu produktu, transport, pensje pracownicze, opłaty medialne, reklama itp. Eksperti uwzględniając wiele czynników w tym tendencję do wzrostów i spadków mają oszacować przyszłoroczne koszty i przychody pojedynczych marketów, z których zostaną obliczone dochody  $D_1, D_2, D_3, D_4$ , które po zsumowaniu dadzą przypuszczalny przyszłoroczny dochód globalny  $D_g$ . Ich ocena nie jest precyzyjna mieści się w różnych przedziałach i aby obliczyć przewidywalny globalny dochód firmy wydaje się sensownym użycie OFN do zobrazowania przychodów i kosztów każdego z marketów podany w milionach zł (tabela1).



Rys. 6 Przewidywane dochody poszczególnych marketów oraz dochód globalny.

Interpretacja dla pierwszej dwójki marketów dla przychodów i kosztów jest następująca:

$l$  - jest wyjściowym punktem dla których prognozowane przychody i koszty mają wzrosnąć;

$(l, l^-)$  - mówi o przewidywanym wzroście przychodów i kosztów. Gdyby koszty mieściły by się w tym przedziale choć niskie według ekspertów mogły by nie starczyć na wszystkie potrzeby. W tym przedziale przychody nie dają zadowalających rezultatów, ale istnieje przypuszczenie, że będą w czasie przynosiły lepsze rezultaty ;

$(l^-, l^+)$  - informuje o spodziewanych optymalnych parametrach przychodów i kosztów

$(l^+, p)$  - mówi o sytuacji przychody w opinii ekspertów będą powodować większe koszty i zmniejszenie się opłacalności produkcji, zaś koszty będą tak duże że opłacalnym było by ich pominięcie;

$p$  - jest to maksymalny poziom kosztów i przychodów jaki może być osiągnięty.

Dla marketów nr: 3 oraz 4 interpretacja tych przedziałów jest inna:

$p$  - w opinii ekspertów jest punktem wyjściowym dla marketów, w których zakładane są zmniejszone koszty i przychody;

$(l^+, p)$  - prognozowane przychody co za tym idzie część kosztów mają zmaleć, jednak bilans ich daje pozytywne perspektywy;

$(l^-, l^+)$  - tak jak dla poprzednich dwóch marketów zawiera informuje o spodziewanych optymalnych parametrach przychodów i kosztów;

$(l, l^-)$  - informuje sytuacji w której większy spadek kosztów daje również duży spadek przychodów co nie jest pozytywną perspektywą;

$p$  - najmniejsze wartości przychodów i kosztów jakie są brane pod uwagę przez ekspertów.

Należy zauważyć że przychody i koszty marketów nr 1 i 2 są liczbami o orientacji pozytywnej zaś pozostałych o orientacji negatywnej (rys. 6). Nie koniecznie ma to wpływ na skierowanie dochodu przykładem staje się sytuacji marketu nr 2, gdzie dochód otrzymany został w postaci OFN skierowanej ujemnie. Choć  $P_2$  i  $K_2$  są dodatnie to koszty rosną szybciej niż przychody co daje spadek dochodu tego w stosunku to punktu jaki obrano za wyjściowy i jest zgodne z intuicją. Można zauważyć również że największy wpływ na  $D_g$  ma market 1, nie tylko ze względu na wielkość kwot na jakich pracuje, ale również ze względu na szerokość nośnika.

Przychody i koszty eksperci starali się dobrać tak by w przyszłym roku otrzymać dochód globalny  $X_g$  w postaci OFN skierowanej pozytywnie. Ważna jest tu interpretacja skierowania jako tendencji wzrostowej dochodu globalnego. Informacja, że  $X_g$  będzie miał tendencję do wzrostu i będzie oscylował w granicach 160 do 194 milionów zł jest więc informacją pozytywną.

Przykład ten pokazuje proste i zgodne z intuicją zastosowanie OFN w modelach ekonomicznych, gdzie istnieje wiele zależności i wpływów zewnętrznych i wewnętrznych na pewne komórki gospodarcze do tego stopnia, iż nawet grupa ekspertów w danej dziedzinie nie jest w stanie dokładnie przewidzieć pewnych sytuacji (kryzys). Wielkim plusem OFN niewielka szerokość nośnika wyniku co umożliwi interpretację bez zbędnych w tym wypadku procedur wyostrzania. Ważną zaletą jest nie tylko zawieranie informacji o sytuacji w danym czasie, ale również pojawienie się parametru umożliwiającego odczytanie tendencji w czasie, które w tym przykładzie musieli uwzględnić eksperci.

#### 4. WNIOSKI I PODSUMOWANIE

Skierowane liczby rozmyte są odpowiedzią na częste zarzuty wytaczane przeciw klasycznym liczbą rozmytym takim jak: zwiększanie rozmytości przy wykonywaniu kolejnych działań, brak możliwości wnioskowania wstecz oraz duże skomplikowanie obliczeniowe. Mają swoją nowatorską algebrę zachowując przy tym całą ideę zbioru rozmytego. Dzięki temu mogą one przechodzić przez wiele operacji nie tracąc przy tym na dokładności. Dają one przy tym możliwość wnioskowania wstecz. Wykonując operację arytmetyczną można otrzymać

liczbę, której nośnik jest nieskończenie mały, interpretuje się ją jako liczbę rzeczywistą. Można więc rozwiązywać zadania, w których część danych jest skierowanymi liczbami rozmytymi a część liczbami rzeczywistymi. Tak jak w arytmetyce liczb rzeczywistych, analogiczne operacje na OFN są przemienne, łączne oraz mnożenie jest rozłączne względem dodawania. Dzięki temu jest się w stanie rozwiązać równania, które nie miało rozwiązania dla klasycznych liczb rozmytych.

Ważnym aspektem w rozważaniach nad skierowanymi liczbami rozmytymi jest obserwacja rozmyta. Istnienie czasu może być wykorzystane jako możliwość wyboru parametru dla przedstawienia krzywej przynależności skierowanej liczby rozmytej w postaci parametrycznej. Czas lub też monotoniczność (narastanie, zanikanie) zjawiska może być dla projektanta systemu decydującym czynnikiem wpływającym na skierowanie liczb.

W przykładach podano kilka możliwości interpretacji skierowania. Rozważane było jako wyznacznik kierunku ruchu, wyznacznik znaku wartości funkcji pierwotnej po czasie oraz pewnej tendencji. Dzięki skierowanym liczbom rozmytym, ekspert może nie tylko oceniać, w jakim stopniu uznaje prawdziwość rozpatrywanego zjawiska, ale także wyrazić swoją ocenę dotyczącą jego dynamiki.

## Literatura

1. Driankov D., Hellendoorn H., Reinfrank M., Wprowadzenie do sterowania rozmytego, Wydawnictwo Naukowo-Techniczne, Warszawa 1996
2. Łachwa A., Rozmyty świat zbiorów, liczb, relacji, faktów reguł i decyzji, Akademicka Oficyna Wydawnicza EXIT Warszawa 2001
3. Kacprzak D., Analiza modelu Leontiewa z użyciem skierowanych liczb rozmytych Politechnika Białostocka Wydział Informatyki Katedra Matematyki [http://212.33.90.158/terw2008/archives/terw2007/resources/articleskacprzak\\_terw2007.pdf](http://212.33.90.158/terw2008/archives/terw2007/resources/articleskacprzak_terw2007.pdf)
4. Kosiński W., On Fuzzy Number Calculus, Int. J. Appl. Math. Comput.Sci., 2006, vol 16, no. 1, pp. 51-57
5. Kosiński W., On fuzzy number calculus Int. J. Appl. Math. Comput. Sci., 2006, vol. 16, no. 1, pp. 51-57
6. Kosiński W., Prokopowicz P., Algebra liczb rozmytych, Matematyka Stosowana, 5 (46), pp. 37-63, 2004
7. Kosiński W., Prokopowicz P., Ślęzak D., (2005), Calculus with fuzzy numbers, Proc. Intern.Workshop on Intelligent Media Communicative Intelligence, Warszawa, September, 2004, L.Bolc, T. Nishida, Z. Michalewicz,(eds), LNCS, Springer, Heidelberg, 2005
8. Piegat A., Modelowanie i sterowanie rozmyte, Akademicka Oficyna Wydawnicza EXIT Warszawa 1999
9. Prokopowicz P., Algorytmizacja działań na liczbach rozmytych i jej zastosowania, rozprawa doktorska, IPPT PAN, 2005
10. Rutkowski L., Metody i techniki sztucznej inteligencji, Inteligencja obliczeniowa, Wydawnictwo Naukowe PWN. Warszawa 2006
11. Rutkowska D., Piliński M., Rutkowski L., Sieci neuronowe, algorytmy genetyczne i systemy rozmyte, Wydawnictwo Naukowe PWN. Warszawa, Łódź 1999
12. Wilczyńska- Sztyma D., Pewne aspekty sterowania w rozumieniu skierowanych liczb rozmytych w środowisku MATLAB, Praca magisterska, napisana pod kierunkiem prof. dra hab. W. Kosińskiego, Bydgoszcz 2007
13. Zadeh L. A.(1965), Fuzzy sets, Information and Control, 8(3), pp. 338-353.
14. Zieliński J.S., Bartkiewicz W. Inteligentne systemy w zarządzaniu. Teoria i praktyka, Pod redakcją Zieliński J.S. Wydawnictwo Naukowe PWN Warszawa 2000

# SYSTEMY MRÓWKOWE W ZASTOSOWANIU DO ROZWIĄZANIA PROBLEMU KOMIWOJAZERA

**Mirosław Modrzejewski**

Uniwersytet Kazimierza Wielkiego  
Instytut Techniki  
II rok MU Edukacja Techniczno-Informatyczna  
ul. Chodkiewicza 30, 85-064 Bydgoszcz  
email: prophetnes@wp.pl

**Streszczenie:** Artykuł porusza dwa zagadnienia. Pierwsze określane jest jako problem komiwojażera popularnie nazywanego TSP (z ang. Traveling Salesman Problem), oraz systemy mrówkowe (z ang. Ant Systems) jako przedstawiciel nowatorskiego podejścia do rozwiązywania problemów optymalizacyjnych z grupy NP-trudnych. Problem TSP jest zagadnieniem optymalizacyjnym polegającym na znalezieniu drogi o najmniejszym koszcie dla wyznaczonej przez komiwojażera trasy. Systemy mrówkowe są to algorytmy wzorujące się na świecie przyrody, a konkretniej na sposobie organizacji kolonii mrówek w poszukiwaniu najkrótszej drogi z mrowiska do pokarmu i z powrotem. Artykuł ma za zadanie zapoznać czytelnika z dwoma zakreślonymi powyżej zagadnieniami, zaprezentować zastosowanie systemów mrówkowych do rozwiązania TSP, zbadać efektywność algorytmów mrówkowych oraz algorytmów klasycznych w poszukiwaniu optimum dla określonych problemów TSP oraz przedstawić otrzymane wyniki wraz z wnioskami końcowymi. Dodatkową częścią artykułu są kierunki dalszych badań, jakie są podejmowane przez naukowców, przy wykorzystaniu filozofii systemów mrówkowych.

**Słowa kluczowe:** Problem komiwojażera, systemy mrówkowe, algorytmy mrówkowe, algorytmy klasyczne, rozwiązanie optymalne, mrówka, jako sztuczny agent, feromon, parowanie feromonu, acotsp, concorde, tsplib, swarm-bots.

## Ant Systems applied to solve the Travelling Salesman Problem

**Abstract:** The paper discusses two problems. The first one is known as the Travelling Salesman Problem (TSP), whereas the second one is defined as the Ant Systems being the representative of innovative attitude to solving optimization problems belonging to the NP-hard group. The TSP problem is an optimizing issue that consists in finding the lowest cost travelling way for the route specified by the travelling salesman. The ant systems are algorithms patterned after the nature, more specifically, after the way an ant colony is organised in order to find the shortest way from the anthill to food and back. The aim of the paper is to familiarize readers with the above two problems, to present application of ant systems to solve the TSP, to examine efficiency of ant algorithms and classical algorithms when searching for the optimum for specific TSP problems as well as to present obtained results together with final conclusions. As an additional part of this paper, the author presented further directions of research undertaken by scientists using philosophy of ant systems.

**Keywords:** The travelling Salesman Problem, ant systems, ant algorithms, classical algorithms, optimal solution, an ant as a false agent, pheromone, pheromone evaporation, acotsp, concorde, tsplib, swarm-bots.

### 1. WSTĘP

Z własnego doświadczenia wiem, że niewiele osób miało styczność z problemem komiwojażera, lub systemami mrówkowymi, a dziwnie brzmiące połączenie tych dwóch terminów, które zawierają się w tytule tego

artykułu powoduje, iż przeciętny odbiorca wpada w lekkie osłupienie dziwiąc się, że nic z przekazanej mu treści nie rozumie. Na pierwszy rzut oka można faktycznie stwierdzić, że temat jest bardzo trudny, jednak rzeczywistość jest zupełnie inna. Charakterystyka obydwu zagadnień, która zarazem prowadzi do zrozumienia samej ich istoty, jest raczej prosta. Tym artykułem chciałbym zachęcić każdego do zapoznania

się z wspomnianymi powyżej zagadnieniami, które mają bardzo duże znaczenie w dzisiejszym świecie, który przez przeciętnego Kowalskiego pojmowany jest w ujęciu globalnym, gdzie wartością jest mobilność, szybkość, skuteczność. Wszystkie te rzeczy ułatwić nam może komiwojażer wspierany przez kolonię mrówek.

## 2. WYJAŚNIENIE ZAGADNIENIA PROBLEMU KOMIWOJAŻERA

Travelling Salesman Problem w skrócie TSP, jest to źródłowa nazwa problemu znanego w Polsce, jako problem wędrującego komiwojażera. Zaliczany jest on do problemów NP – trudnych do rozwiązania z powodu dużego skomplikowania w obliczeniach, które mogą trwać nawet wiele lat. Główną ideę samego zagadnienia można sformułować następująco:

Przyjmij liczbę miast i podaj koszt podróży z dowolnego miasta do innego dowolnego miasta, a następnie określ, jaki jest najmniejszy koszt, oraz najkrótsza droga tak, aby każde miasto zostało odwiedzone tylko raz i zakładając, że miastem ostatnim jest miasto z którego wyruszyliśmy.

Zależy nam na tym, aby wyznaczyć przez nas trasę pokonać przy jak najmniejszym koszcie, która określana jest terminem trasy optymalnej. Koszt takiej podróży może być rozumiany na wiele sposobów jak np.: 1) najkrótsza droga z miasta A do miasta docelowego B, 2) jak najmniejszy koszt spalania benzyny dla danej trasy, który może być uzyskany nie tylko w przypadku trasy najkrótszej, ale na przykład trasy dłuższej bez korków ulicznych. Są to tylko dwa proste przykłady, które zawierają się w samym zagadnieniu TSP.

TSP ma bardzo długą historię. Jego początki według Z. Michalewicza oraz D.B. Fogel [2] sięgają roku 1759, gdzie problem został wspomniany przez wybitnego w tamtych czasach szwajcarskiego matematyka i fizyka Leonarda Eulera. W tamtym okresie problem ten nie był rozpoznawany jako TSP, występował on pod inną nazwą. Problem dotyczył trasy konika szachowego, a dokładnie polegał on na podaniu takiej trasy konika, w której odwiedzi on wszystkie 64 pola szachownicy dokładnie raz [2].

Matematyczne problemy związane z TSP zostały również poruszone w 1800 roku przez dwóch naukowców: szkockiego matematyka Ser Williama Hamiltona oraz brytyjskiego matematyka Thomasa Kirkmana. Rozważania na temat wczesnych prac Hamiltona i Kirkmana, oraz Eulera możemy znaleźć w pracy Graph Teory 1736-1936. W tej pracy przedstawiono ciekawy problem miasteczka Königsberg (Królewiec), w którym jak to zartobliwie napisano:

mieszkańcy jako rozrywkę próbują odnaleźć trasę dookoła miasta w taki sposób, aby przekroczyć każdy z siedmiu mostów tylko raz [3]. W polskiej literaturze problem ten jest znany pod nazwą problemu mostów królewieckich. Według wspomnianej pracy mieszkańcy miasteczka próbowali znaleźć rozwiązanie tego problemu, jednak jak się ostatecznie okazało bez skutku. Próby rozwiązań były dokonywane przed rokiem 1730, a wiec można stwierdzić, że pewna forma TSP była już znana przed 1759 rokiem, nie jak podaje książka Michalewicza oraz Fogel. W 1736 roku słynny matematyk tamtych czasów Euler zgłębił problem, oraz zaproponował pewien algorytm. Ostatecznie okazało się jednak, że rozwiązanie tego problemu nie jest możliwe ze względu na nieparzystą liczbę wyjść mostów zarówno na oba brzegi rzeki jak i na wyspy [4].

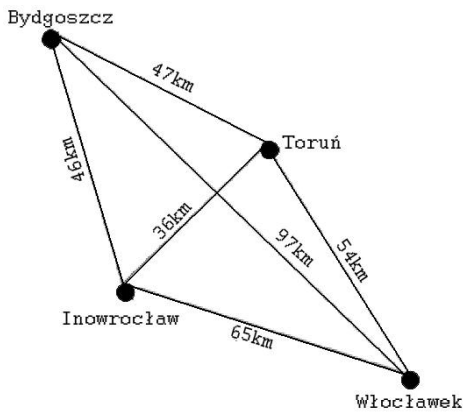
Początki TSP były trudne, jak każda nowa nauka. W latach 20, dwudziestego wieku, matematyk i ekonomista Karl Menger przedstawił to zagadnienie swoim kolegom na swojej uczelni w Vienna. Pierwsze konkretne nazewnictwo problemu TSP terminem „komiwojażer” nastąpiło w latach 30 dwudziestego wieku w Vienna i na Harvardzie przez niemieckiego naukowca Karl Mengera. W późnych latach 30, problem ponownie pojawił się w matematycznych kręgach uczelni Princeton. W latach 40, problem był studiowany przez statystyków i matematyka Merill Flood, którzy to spopularyzowali problem wśród swoich kolegów w korporacji RAND. Ostatecznie, problem TSP zyskał zły rozgłos, jako rodzaj problemu bardzo trudnego do rozwiązania. Sprawdzanie ścieżek jedna po drugiej nie było brane pod uwagę z powodu zbyt dużej liczby możliwych rozwiązań, innych alternatywnych propozycji rozwiązań tego problemu nieznanego w tamtych czasach. Wysokie skomplikowanie problemu przedstawia poniższa tabela, która pokazuje złożoność problemu, rozumiana, jako liczba wszystkich możliwości do rozpatrzenia w stosunku do podwojonej liczby miast, tzn.  $n!/(2n) = (n-1)!/2$ .

Tabela 1. Wielkość problemu [5]

Miast	Wielkość problemu
3	1
5	12
7	360
10	181440
15	43589145600
20	60822550204416000
25	3102242008662000000000
30	442088099686985000000000000000

W późniejszym czasie problem był rozpowszechniany przez Hassler Whitney oraz Merrill M. Flood na uczelni Princeton. Osiągnięcia K. Mengersera oraz H. Whitney'ego zostały umieszczone w opracowaniu A. Schrijver On the history of combinatorial optimization, które zawiera historię problemów optymalizacyjnych, dotyczących również TSP do lat 60.

Aby zwizualizować ten problem na bazie naszych realiów poniżej można zobaczyć rysunek przedstawiający problem czterech miast województwa Kujawsko-Pomorskiego. Zadaniem jest znalezienie jak najkrótszej trasy, przy założeniu, że miastem startowym jest Inowrocław, które przedstawiają kombinację wszystkich możliwych rozwiązań danego problemu.



Rys. 9 Graf ważony przedstawiający cztery polskie miasta

Całkowitą liczbę wszystkich możliwych rozwiązań powyżej przedstawionego przykładu, który reprezentuje zagadnienie TSP, można scharakteryzować, jako cykl Hamiltona, który jest reprezentowany wzorem:

$$\frac{1}{2} \cdot (n - 1)! \rightarrow n > 2 \tag{1}$$

Za pomocą tego wzoru można obliczyć wielkość problemu dla dowolnej liczby miast, co czyni zagadnienie TSP bardzo trudnym do rozwiązania. Dla naszego przykładu czterech miast całkowita ilość możliwych rozwiązań jest równa  $\frac{1}{2} \cdot 3!$ , czyli 3 rzeczywiste rozwiązania dla 6 dostępnych możliwych tras. Dużą zaletą jest to, iż wzór na cykl Hamiltona wyklucza powtórzenie się tego samego rozwiązania. Za pomocą tego wzoru można obliczyć wielkość problemu dla dowolnej liczby miast, co czyni zagadnienie TSP bardzo trudnym do rozwiązania. Dla

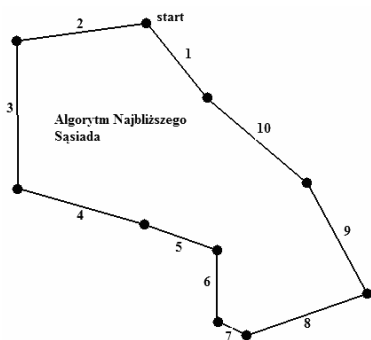
naszego przykładu czterech miast całkowita ilość możliwych rozwiązań jest równa  $\frac{1}{2} \cdot 3!$ , czyli 3 rzeczywiste rozwiązania dla 6 dostępnych możliwych tras. Dużą zaletą jest to, iż wzór na cykl Hamiltona wyklucza powtórzenie się tego samego rozwiązania.

Tabela 2 Możliwe rozwiązania cyklu Hamiltona dla 4 przyjętych miast

Trasa	Dystans [km]
<b>Inowr., Bydgoszcz, Toruń, Włocławek, Inowr.</b>	<b>212</b>
<b>Inowr., Włocławek, Toruń, Bydgoszcz, Inowr.</b>	<b>212</b>
<b>Inowr., Bydgoszcz, Włocławek, Toruń, Inowr.</b>	<b>233</b>
<b>Inowr., Toruń, Włocławek, Bydgoszcz, Inowr.</b>	<b>233</b>
<b>Inowr., Toruń, Bydgoszcz, Włocławek, Inowr.</b>	<b>245</b>
<b>Inowr., Włocławek, Bydgoszcz, Toruń, Inowr.</b>	<b>245</b>

Tak jak można łatwo zauważyć tabela numer 2 prezentuje wszystkie możliwe rozwiązania. Cykl Hamiltona danego grafu, po którym się poruszamy, mówi nam jednoznacznie ile jest możliwych dostępnych rozwiązań pokonania założonej drogi. Jest to informacja wyjściowa do podania optymalnego rozwiązania TSP.

Tak jak wspomniano na samym początku sama charakterystyka problemu jest stosunkowo prosta, natomiast samo rozwiązanie niestety już nie. Aby sprostać problemowi TSP, wymyślono wiele różnych algorytmów, określanych dzisiaj terminem klasyczne, za pomocą których możliwe jest poszukiwanie rozwiązań optymalnych lub rozwiązań bliskich optymalnym. Rozróżniono dwie główne grupy algorytmów, do których zaliczamy: 1) algorytmy znajdujące dokładne rozwiązanie (algorytmy czasochłonne) oraz 2) algorytmy nie w pełni optymalne nazywane heurystycznymi. Pierwsza grupa są to algorytmy, które sprawdzają się tylko dla małych problemów. Do przykładowych algorytmów możemy zaliczyć m.in. *branch-and-bound*, *branch-and-cut*. Druga grupa to algorytmy znajdujące rozwiązanie bliskie optymalnemu w stosunkowo krótkim czasie. Sprawdzają się one nawet dla dużych problemów. Do przykładowych algorytmów możemy zaliczyć m.in. algorytm najbliższego sąsiada (z ang. *Nearest Neighbour*), algorytm zachłanny (z ang. *Greedy algorithm*), algorytm drugiego przybliżenia 2-opt, k-opt, jak i również najskuteczniejszy ze znanych w dzisiejszej dobie algorytmów klasycznych Lin-Keringhana.



Rys. 10 Metoda najbliższego sąsiedztwa

Tak jak można łatwo zauważyć, TSP jest bardzo ważnym zagadnieniem, którego zastosowanie może ułatwić ludziom życie. W dzisiejszej dobie coraz popularniejsze są geolokalizatory z funkcją znajdowania najkrótszej trasy, takie jak np. map24 z portalu interia lub choćby docel z wp.pl. Bez wątplenia można stwierdzić, iż każdy podróżujący niejednokrotnie sprawdził w jednym z podanych powyżej serwisów, trasę, jaką mógłby pojechać, aby zaoszczędzić choćby przysłowiową złotówkę. Choć ich działanie pozostawia wiele do życzenia, to ciągle prace nad doskonaleniem zagadnienia TSP mogą nam dać świetny rezultat w postaci dobrej optymalnej trasy do miejsca naszych wymarzonych wakacji.

### 3. WYJAŚNIENIE ZAGADNIENIA SYSTEMÓW MRÓWKOWYCH

Pracowita mrówka, niewielkie stworzenie, które może dokonać wielkich rzeczy. Czy aby na pewno tak jest? Jak się okazuje nie do końca. Poniższy cytat w bardzo dobry sposób prezentuje to zagadnienie:

*A single ant or bee isn't smart, but their colonies are. The study of swarm intelligence is providing insights that can help humans manage complex systems, from truck routing to military robots [6].*

Pojedyncza mrówka pomimo swej prostoty, jako jednostka wraz z całą społecznością mrówek potrafi prezentować bardzo wysokie zorganizowanie w pracy. Jako kolonia mrówek, mogą dokonać rzeczy, które przekraczają zdolność pojedynczego osobnika. Pojęcie systemów mrówkowych można zdefiniować, jako system informatyczny, który realizuje algorytmy równoległe. Działają one na analogicznych zasadach, co odpowiadające im procesy w świecie rzeczywistym. Wiele z nich bazuje na procesach dobrze poznanych przez zoologów, czy fizyków molekularnych.

Systemy mrówkowe to pojęcie wydawałoby się zagadkowe nie mające za dużo wspólnego z informatyką i algorytmami, a jednak jest wręcz przeciwnie. Tak naprawdę tego typu podejście nie jest nam obce w dzisiejszych czasach. Wielu naukowców wzoruje się na naszej matce naturze jak na obiekcie natchnienia, wzorca dobrego do naśladowania. Nie jest to sytuacja dziwna, w końcu matka natura miała miliony lat, aby wykształcić metody rozwiązań wielu problemów. Powracając do sedna sprawy, jak sama nazwa wskazuje w tym zagadnieniu uczestniczą mrówki, dobrze zorganizowane, pracowite organizmy, potrafiące o dziwo i tu uwaga „znaleźć optymalne rozwiązanie dla określonej drogi”. Droga z punktu widzenia mrówek jest trasa prowadząca od gniazda mrówek do pożywienia. Jak to się zdarza w naturze i w jej wytworach, często na drodze takiej mrówki staje przeszkoda. Najprostszym sposobem prezentacji zagadnienia systemów mrówkowych będzie przedstawienie mikroświata w postaci gniazda, pożywienia, przeszkody oddzielającej gniazdo od pożywienia oraz dwóch dróg prowadzących do pożywienia. Celem mrówek jest dojście do pożywienia, oraz powrót do gniazda w jak najszybszym czasie, czyli wybranie trasy najkrótszej. Z obserwacji mrówek wiemy, że po pewnym czasie znajdują one krótszą z dwóch dostępnych tras. Należy sobie zadać pytanie: jak to jest możliwe? Mrówki wydzielają pewną substancję, która nazywana jest feromonem. Wędrując od gniazda do miejsca pokarmu mrówki zostawiają tzw. ślad feromonowy

Rzeczą istotną jest to, iż pozostawiony ślad feromonowy jest intensywniejszy na drodze częściej uczęszczanej. W tym momencie można zauważyć sposób, w jaki mrówki odnajdują najkrótszą trasę. Kierują się one intensywnością feromonu, im intensywniejszy tym droga częściej uczęszczana, im częściej uczęszczana oznacza to, że do pokarmu jest bliżej, a bliżej oznacza lepiej i jest rozwiązaniem optymalnym. Zobaczmy, jak to wygląda na rysunku z dwoma dostępnymi drogami. Tak jak można zauważyć mamy pokazaną drogę mrówek z gniazda do pokarmu. Do pokarmu prowadzą dwie drogi, które powstały za przyczyną głazu.



Rys. 11 Dwie drogi do pokarmu

Obydwie drogi są uczęszczane w dwóch kierunkach, jako że mrówki idą po pokarm, a następnie wracają z nim do mrowiska. Na pierwszy rzut oka można zauważyć, która z dróg jest krótsza, a która dłuższa, jednak mrówki tego nie wiedzą. Tak jak już wspomniałem powyżej, im droga dłuższa, tym jest mniej uczęszczana z powodu mniejszej intensywności feromonu, im droga krótsza tym częściej uczęszczana i intensywność feromonu jest wyższa.

Idąc dalej tym tropem, należałoby sobie zadać pytanie: kiedy mrówki dojdą do tego, którą drogą należy iść, aby dotrzeć i wrócić szybciej? Wszystko to zależy od ilości mrówek, jakie wybrały określoną drogę, oraz długości trasy. Na początku mrówki losowo wybierają drogę. Oczywiście jest, że krótszą drogą mrówki szybciej wrócą z pokarmem, co spowoduje szybsze nanoszenie feromonu i wzrost jego intensywności. Jednak tak naprawdę nie da się określić, kiedy zostanie wybrana optymalna trasa. Na pewno można stwierdzić, że prędzej czy później zostanie ona wybrana. Szybkość w dużej mierze będzie zależała od szczęścia. Mam tu na myśli sytuację losową, jeśli mrówek będzie więcej i dobrze wybiorą one trasę na samym początku, znalezienie optymalnej drogi zajmie mniej czasu, natomiast gdy tych mrówek będzie mniej, znalezienie rozwiązania optymalnego w krótkim czasie będzie mało prawdopodobne.

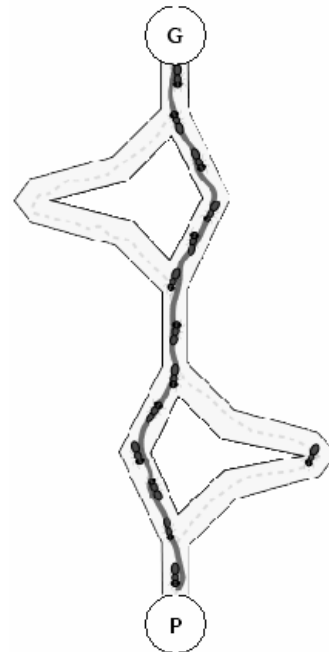
Pojęcie algorytmów mrówkowych zostało zdefiniowane przez Marco Dorigo w 1992 roku, jako wielo-agentowy system zainspirowany przez obserwację zachowań prawdziwych kolonii mrówek wykorzystujący zagadnienie *stigmergy* (STY) (pośrednia komunikacja). Dziedzina „algorytmów mrówkowych” wywodzi się z obserwacji naturalnego środowiska życia mrówek. Pojęcie STY pełni ważną rolę w algorytmach mrówkowych, ponieważ dzięki implementacji tzw. zmiennych STY możliwe jest ich funkcjonowanie. Zmienne te zawierają informację używane przez sztuczne mrówki do komunikacji pośredniej. Dla przykładu, w algorytmie ACO zmienne STY są specjalnie zdefiniowane i używane przez mrówki do zmiany sposobu budowania rozwiązania dla rozważanego problemu.

**Tabela 3** Algorytmy ACO w chronologicznej kolejności ich pojawienia się [7]

Algorytmy ACO	Dla TSP	Twórcy
System Mrówkowy AS	TAK	Dorigo 1992
Elitarny System Mrówkowy	TAK	Dorigo 1992

Ant-Q	TAK	Gambardella i Dorigo 1995-96
System Kolonii Mrówek ACS	TAK	Dorigo i Gambardella 1997
Max-Min AS	TAK	Stützle i Hoos 1996, 2000
System oceny AS <sub>rank</sub>	TAK	Bullnheimer, Hartl i Strauss 1997-99
ANTS	NIE	Maniezo 1999
Hoyer Cube AS	NIE	Blum Roli i Dorigo 2001, Blum i Dorigo 2004

Ten model został potraktowany, jako źródło inspiracji do zaprojektowania nowatorskich algorytmów. Algorytmy były tworzone z myślą zastosowania ich do problemów optymalizacji i problemów kontroli zaburzeń. Wdrożonym pojęciem wykorzystywanym przy algorytmach mrówkowych jest sztuczny agent (z ang. *artificial agent*), który jest odpowiednikiem zwykłej mrówki. Sztuczne mrówki w rozumieniu programistycznych razem współpracują w celu rozwiązywania problemów obliczeniowych. Poniżej możemy zobaczyć wykaz algorytmów mrówkowych wraz z datą ich ukazania się.



**Rys. 12** Optymalna trasa mrówek

Optymalizacja kolonią mrówek (z ang. *Ant Colony Optimization*, ACO) – jest jednym z najbardziej znanych w obecnych czasach algorytmów mrówkowych. Został

on po raz pierwszy zdefiniowany przez Marco Dorego, Di Caro oraz Gambardella w 1999 roku jako technika dla rozwiązywania problemów dyskretnej optymalizacji. ACO został przedstawiony, jako algorytm, który może znaleźć dobrą ścieżkę poprzez graf. Jest to algorytm inspirowany teorią *foragingu* [8] dla kolonii mrówek jak i problemów dyskretnej optymalizacji. Algorytm jest przeznaczony do rozwiązywania dwóch rodzajów problemów: a) optymalizacyjnych statycznych, czyli problemów gdzie podczas pracy żadne dane się nie zmieniają i tu przykładem może być TSP, jak i b) problemów dynamicznych, gdzie informacje podczas działania programu zmieniają się i tu przykładem może być problem wyznaczania trasy, w którym parametrem dynamicznym jest ciągle zmieniający się ruch. Podstawą do rozpoczęcia badań nad algorytmami mrówkowymi było samo obserwowanie środowiska mrówek. Zaobserwowaną ciekawostką było to, iż w przeważający sposób mrówki porozumiewają się ze sobą za pomocą substancji chemicznych przez nie produkowanych. Jak już wspomniałem bardzo ważną rzeczą w tym algorytmie jest pośrednia komunikacja *foraging*, reprezentowana przez ślad feromonowy. Parowanie tego feromonu ma tą przewagę, że może zapobiegać konwergencji (zbieżności) dla optymalnych lokalnych rozwiązań. Gdybyśmy założyli, że nie istnieje zagadnienie parowania, to za każdym razem każda ze ścieżek wybierana przez pierwszych sztucznych agentów była by traktowana jednakowo, byłaby jednakowo atrakcyjna, co nie miałyby zastosowania do rozwiązywania problemów optymalizacyjnych. W ten sposób, kiedy jedna mrówka znajdzie dobrą ścieżkę z kolonii do źródła pokarmu, pozostałe mrówki przychylają się bardziej do tej trasy. Pomysłem dla algorytmu ACO jest naśladowanie tego zachowania za pomocą sztucznych agentów poruszających się w obrębie grafu w celu rozwiązania danego problemu. Algorytm ACO został użyty do rozwiązania problemu TSP. Algorytm ten ma przewagę nad algorytmami genetycznymi lub algorytmem symulowanego wyżarzania. Ważną jego cechą jest to, że dla grafu zmieniającego się dynamicznie, algorytm ACO może pracować ciągle i zaadaptować się do zmian w czasie rzeczywistym. Zasadą takiego działania jest metoda wyznaczania trasy w sieci (z ang. *network routing*) i systemy miejskiego transportu (z ang. *urban transportation systems*). Poniżej możemy zobaczyć pseudo kod algorytmu ACO:

Kod 1 Pseudo Kod algorytmu ACO [9]

```

1  procedure ACO_MetaHeuristic
2      while(not_termination
3          generateSolutions()
4          pheromoneUpdate()

```

```

5      daemonActions()
6      end while
7  end procedure

```

Gdzie poszczególne funkcje mają poniższe zadania.

**generateSolutions()** – zarządza kolonią mrówek, które jednocześnie i asynchronicznie odwiedzają sąsiadujące stany danego rozważanego problemu przez przesuwanie się poprzez sąsiednie wierzchołki, opisanego problemu, danego grafu G. Przesuwanie mrówek odbywa się przy pomocy stochastycznie lokalnie podejmowanych decyzji, które są podejmowane na bazie śladów feromonowych i innych informacji heurystycznych. W ten sposób mrówki budują w sposób przyrostowy rozwiązanie i oceniają cząstkowe rozwiązania, które będą wykorzystane przez funkcję *pheromoneUpdate()*, aby zdecydować ile feromonu zostanie pozostawione,

**pheromoneUpdate()** – jest to funkcja, za pomocą której modyfikowany jest ślad feromonowy. Ten ślad może wzrastać, dzięki pracy mrówek, które pokonują kolejne krawędzie (wierzchołki), jak i może się zmniejszać, przez naturalne zaimplementowane parowanie. Parowanie feromonu zapobiega skupieniu się na jednym obszarze przeszukiwań, a zarazem zachęca do przeszukiwania nowych do tej pory nieodwiedzanych miejsc,

**daemonActions()** – jest to funkcja, która jest używana do implementacji zcentralizowanych działań, które nie mogą być przeprowadzone przez pojedynczą mrówkę. Przykładem działania tej funkcji jest włączenie procedury lokalnej optymalizacji, lub zbieranie globalnych informacji, które mogą być wykorzystane do podejmowania decyzji czy użytecznym jest, lub też nie pozostawienie dodatkowego feromonu, aby wpłynąć na proces przeszukiwania z poza lokalnej perspektywy.

Do dwóch najważniejszych wzorów, które opisują zagadnienie systemów mrówkowych należą:

**Wybór ścieżki** – mrówka przebędzie drogę z punktu *i* do punktu *j* z prawdopodobieństwem równym:

$$p_{ij} = \frac{\tau_{i,j}^{\alpha} \cdot \eta_{i,j}^{\beta}}{\sum \tau_{i,j}^{\alpha} \cdot \eta_{i,j}^{\beta}} \quad (2)$$

gdzie

$\tau_{i,j}$  - ilość feromonu na ścieżce *i,j*,

$\alpha$  - parametr do kontroli wpływu  $\tau_{i,j}$ ,

$\eta_{i,j}$  - określa atrakcyjność ścieżki *i,j*,

$\beta$  - parametr do kontroli wpływu  $\eta_{i,j}$ .

**Uaktualnienie feromonu** – to zagadnienie reprezentowane jest wzorem

$$\tau_{i,j}^{(n)} = p\tau_{i,j}^{(s)} + \Delta\tau_{i,j} \quad (3)$$

gdzie wskaźnik  $n$  oznacza normę, zaś  $s$  – stan, ilość fermonu

$\tau_{i,j}$  - ilość feromonu na ścieżce  $i,j$ ,

$p$  – skala parowania feromonu,

$\Delta\tau_{i,j}$  - oznacza ilość feromonu pozostawionego, określane wzorem

Poniżej można zobaczyć bardziej szczegółowy pseudo kod jednego z wielu algorytmów mrówkowych, określanego jako ACS (z ang. *Ant Colony System*), czyli optymalizacja kolonia mrówek.

Kod 2 Pseudo kod algorytmu kolonii mrówek ACS [10]

```

1 Initialize
2 Repeat {
3   Place each ant in a randomly
  chosen city;
4   For each ant
5     Repeat {
6       Choose NextCity (each ant);
7       Update pheromone levels using
  a local rule;
8     } Until (No more cities to
  visit);
9     Return to the initial cities;
10    Compute the length of the Tour
  found by each ant;
11  End For;
12  Update pheromone level using a
  global rule;
13 }
14 Print Best Path;
```

Podsumowując można zatem wyróżnić kilka etapów cyklu życia mrówki w systemie poszukującym optymalnego rozwiązania:

- Wybór losowego miasta.
- Miasto zostało wybrane.
- Konstruowanie rozwiązania. Wybór kolejnego miasta.
- Powrót do miasta startowego.
- Po zakończeniu przez wszystkie mrówki swoich tras następuje pozostawienie feromonu na ścieżce, którą przebyły.

#### 4. ZASTOSOWANIE SYSTEMÓW MRÓWKOWYCH DO ROZWIĄZANIA PROBLEMU KOMIWOJAŻERA

Drogi czytelniku, posiadasz już podstawową wiedzę z zakresu problemu TSP oraz systemów mrówkowych (SM). Najwyższy już czas, aby połączyć te dwa zagadnienia i przedstawić rezultaty zastosowania kolonii mrówek w odnajdowaniu optymalnej trasy. Celem tej części artykułu jest przedstawienie zależności pomiędzy tymi dwoma zagadnieniami, aby zrozumieć przyczynę zastosowania SM do TSP, oraz innych łączących je elementów.

TSP jest problemem bardzo starym. Doczekał on się wielu badań, znaleziono wiele rozwiązań, mniej lub bardziej efektywnych, poświęcono mu bardzo dużo miejsca w literaturze. Z punktu widzenia zdobytej wiedzy na ten temat można powiedzieć, że mrówki zyskały dużo korzyści dzięki TSP, a TSP zyskało również dużo korzyści dzięki mrówką. TSP pełniło bardzo dużą rolę w algorytmie optymalizacji kolonią mrówek (ACO). TSP było bardzo ważne, ponieważ jako problem NP-trudny, posłużył on do pierwszych testów funkcjonowania systemów mrówkowych (AS). Sami autorzy wymieniają kilka przyczyn, dla których zdecydowali się wybrać ten, a nie inny problem do przedstawienia funkcjonowania algorytmu ACO. Do takich powodów zostały zaliczone m.in.:

TSP jako problem NP-trudny ma szerokie zastosowanie, W dosyć łatwy sposób można zaimplementować algorytm ACO do tego problemu, Kod jest łatwy do zrozumienia, dzięki TSP samo serce ACO nie jest zasłanianie dodatkowymi technicznymi sprawami, które mogłyby skomplikować sam kod i zmniejszyć przejrzystość ACO.

Tak jak powyżej wspomniałem pierwszy system mrówkowy był sprawdzany na problemie TSP. Początkowo wyniki testów AS na bazie TSP nie były zbyt obiecujące, jednak po upływie pewnego czasu sama filozofia wykorzystania mrówek zaczęła się rozwijać i powstawały kolejne rozszerzenia AS, mam tu na myśli m.in. Elitist Ant Systems, MMAS, czy AS<sub>rank</sub>. Wszystkie te rozwinięcia pierwszego bazowego AS, spowodowały bardzo duże usprawnienie systemu i wzrost jego wydajności. Na chwilę obecną algorytmy ACO uznawane są za jedno z najbardziej skutecznych, efektywnych algorytmów zajmujących się problemami optymalizacyjnymi, które rozwiązują problem TSP, oraz wiele innych trudnych problemów. Jeśli chodzi o różnice pomiędzy bazowym AS a rozszerzeniami to dotyczą one

sposobu uaktualniania śladu feromonowego, jak i również pewne dodatkowe rzeczy związane z zarządzaniem tymi śladami. Oprócz algorytmów, rozszerzeń AS, które nie wniosły dużych zmian, w późniejszym okresie zaproponowano również algorytmy, które wprowadziły większe zmiany. Do tych algorytmów możemy zaliczyć systemy kolonii mrówek (ACS), Ant-Q, ANTS (z ang. *Approximate Nondeterministic Tree Search*) wykorzystujące techniki dolnej granicy w programowaniu, czy środowisko hyper-cube dla ACO.

Jeszcze jedną rzeczą, o której należy wspomnieć, jest to zastosowanie algorytmów lokalnego przeszukiwania, które w znaczący sposób przyspieszyły jego działanie i pozwoliły na osiągnięcie lepszych rezultatów w rozwiązaniach TSP. Do stosowanych algorytmów zaliczane są: 2-opt, 2,5-opt jak i najskuteczniejszy 3-opt. W książce *Ant Colony Optimization* Dorigo i Stützle można znaleźć wiele badań dotyczących tego zagadnienia, pokazujących skuteczność wszystkich trzech sposobów przeszukiwania w oparciu o dwa wybrane problemy z bazy TSPLIB.

Tak jak można zauważyć TSP był pierwszym problemem kombinatorycznej optymalizacji, który posłużył do sprawdzania skuteczności funkcjonowania algorytmów ACO. Pierwszym testowanym algorytmem był AS, który uzyskał dobre rezultaty, ale tylko dla problemów TSP o małej wielkości. Przy problemach o większej ilości miast wyniki były dość słabe. Głównym celem zastosowania AS dla TSP było potwierdzenie przez Dorigo koncepcji zastosowania tego typu algorytmów do rozwiązywania problemów optymalizacyjnych. Na chwilę obecną dostępna jest dość duża gama algorytmów ACO. Można powiedzieć, iż te algorytmy wniosły duży wkład w rozwój rozwiązywania trudnych problemów i osiągnęły bardzo dobre rezultaty niejednokrotnie znajdując najlepsze rozwiązanie danego problemu.

## 5. BADANIE EFEKTYWNOŚCI ALGORYTMÓW

W tej części zostanie przedstawiony sposób przeprowadzenia części eksperymentalnej, której celem jest dokonanie porównania efektywności algorytmów mrówkowych w stosunku do algorytmów klasycznych. Zostanie dokonane porównanie efektywności oprogramowania ACOTSP oraz CONCORE, jako dwóch programów mogących poradzić sobie z problemem NP – trudnym, jakim jest TSP. Będzie to również konfrontacja algorytmów klasycznych z całkowicie nowym podejściem, jakim jest naśladowanie koloni mrówek do rozwiązania problemów optymalizacyjnych.

Testy badania skuteczności rozwiązań zostały przeprowadzone na komputerze klasy PC o ogólnej specyfikacji:

- Procesor: AMD Athlon Dwurdzeniowy 2 x 2,6 GHz,
- Pamięć RAM: 2 GB,
- System operacyjny: Windows Vista Premium.

### 5.1. Charakterystyka danych wejściowych

Program CONCORDE nie potrzebuje żadnych dodatkowych parametrów, aby znaleźć rozwiązanie problemu. Jeśli chodzi o oprogramowanie ACOTSP, parametry te muszą być odpowiednio dobrane, aby rezultaty były możliwie najlepsze. Zgodnie z plikiem readme i według zaleceń autora dokumentu każdy z tych systemów uzyskuje dobre rezultaty dla innych parametrów. Nie zawsze ustawienia domyślne dają dobre rezultaty. Do najważniejszych parametrów, które mają wpływ na końcowy rezultat programu zaliczamy:

Tabela 4 Wyróżnienie głównych parametrów mających wpływ na końcowe rozwiązanie problemu

L.p.	Opis parametru	Komenda
1	Liczba uczestniczących mrówek	-m / --ants
2	Parametr określający wpływ śladu feromonowego	-a / --alpha
3	Parametr określający wpływ heurystycznych informacji	-b / --beta
4	Parametr określający parowanie feromonu	-e / -rho
5	Prawdopodobieństwo najlepszego wyboru w konstruowaniu trasy	-q / --q0
6	Wybór lokalnego przeszukiwania 0: brak, 1: 2-opt, 2: 2.5-opt, 3: 3-opt	-l / -- localsearch

Dla wymienionych powyżej parametrów starano się znaleźć jak najlepszą kombinację tak, aby otrzymane rozwiązanie był jak najlepsze. Testy zostały przeprowadzone dla problemu wielkości 783 miast o oznaczeniu rat783, dla którego optimum jest równe 8806. Przeprowadzono wiele testów dla każdego rodzaju algorytmu mrówkowego w celu ustalenia optymalnych parametrów, dzięki którym wynik będzie jak najbliższy rozwiązaniu najlepszemu.

## 5.2. Sposób przeprowadzenia eksperymentu

Dzięki ustalonym parametrom możemy przejść do porównania algorytmów klasycznych z algorytmami mrówkowymi. Ostatnią rzeczą, jaką nam pozostaje ustalić jest wybór problemów z biblioteki TSPLIB oraz ustalenie ilości tych problemów. Ustalono, iż 10 wybranych problemów pozwoli w znaczący sposób zbadać efektywności obydwóch typów algorytmów. Będą one badane według poniższych zasad:

Jeśli chodzi o ACOTSP, to dla każdego problemu program będzie uruchamiany trzykrotnie dla  $r = 10$ , natomiast CONCORDE trzykrotnie bez dodatkowych parametrów, Skuteczność danego algorytmu będzie oceniana następująco:

przez podanie wyniku otrzymanej długości trasy, procentowo w postaci: optimum spełnione w  $x\%$ , tak jak to zostało przedstawione w tabeli numer 5,

Po zebraniu wszystkich wyników, zostanie sporządzony ogólny wykres efektywności wszystkich algorytmów. Dla każdego algorytmu zostanie zsumowana wartość „optimum spełnione w  $x\%$ ”, przez co otrzymamy całkowitą ilość punktów na 1000 możliwych. Ten sposób prezentacji danych pomoże w łatwy sposób ocenić, jaka jest hierarchia wszystkich testowanych algorytmów na przekroju 10 zbadanych próbek.

Poniżej w tabeli zostało przedstawione 10 problemów, które zostaną poddane testom wraz z oczekiwaną wartością optimum, która będzie docelowa dla testowanych algorytmów.

**Tabela 7** Lista rozpatrywanych problemów wraz z podanym optimum

Lp	Kod problemu	Opt.	Opis
1	<b>Eil51</b> (Christofides, Eilon)	426	Problem dla 51 miast
2	<b>D198</b> (Reinelt)	15780	Reprezentuje problem wywierconych dziur w materiale (z ang. <i>Dribling Problem</i> ). Wielkość problemu 198 dziur
3	<b>Gil262</b> (Millet, Johnson)	2378	Problem dla 262 miast
4	<b>Lin318</b> (Lin, Kernighan)	42029	Problem dla 318 miast
5	<b>Pcb442</b> (Groetschel, Juenger, Reinelt)	50778	Reprezentuje problem wywierconych dziur w materiale (z ang. <i>Dribling Problem</i> ). Wielkość problemu

442 dziury			
6	<b>Rat783</b> (Pulleyblank)	8806	Problem połączonej sieci elektrycznej dla 783 punktów
7	<b>Pcb1173</b> (Juenger, Reinelt)	56892	Reprezentuje problem wywierconych dziur w materiale (z ang. <i>Dribling Problem</i> ). Wielkość problemu 1173 dziur
8	<b>D1291</b> (Reinelt)	50801	Reprezentuje problem wywierconych dziur w materiale (z ang. <i>Dribling Problem</i> ). Wielkość problemu 1291 dziur
9	<b>Nrw1379</b> (Bachem, Wottawa)	56638	Problem 1379 miejscowości w Nadreni Północna-Westfalia
10	<b>Pr2392</b> (Padberg, Rinaldi)	378032	Problem 2392 miast

## 5.3. Wnioski i podsumowanie

Po przeprowadzeniu szeregu testów zgodnie z ustalonymi zasadami z całą pewnością można stwierdzić wyższość algorytmów mrówkowych (ALM) nad algorytmami klasycznymi (ALK). Z dziesięciu badanych próbek tylko w jednym przypadku doszło do sytuacji, w której ALK, czyli algorytm Lin-Kernighan (LK) uzyskał lepszy rezultat od wszystkich pozostałych ALM. Mogło to wynikać z charakterystyki problemu, jakim jest nrw1379. W pozostałych przypadkach algorytm LK wyprzedził w najlepszym przypadku i tak stare już ALM, czyli AS oraz EAS, których początek sięga 1992 roku. Pozostałe algorytmy z grupy ALK zdecydowanie są poza czołówką, do której zaliczamy wszystkie ALM oraz algorytm LK. Spełniały one optimum w zakresie od 79% do 89%, co jest wynikiem dalekim od najlepszych.

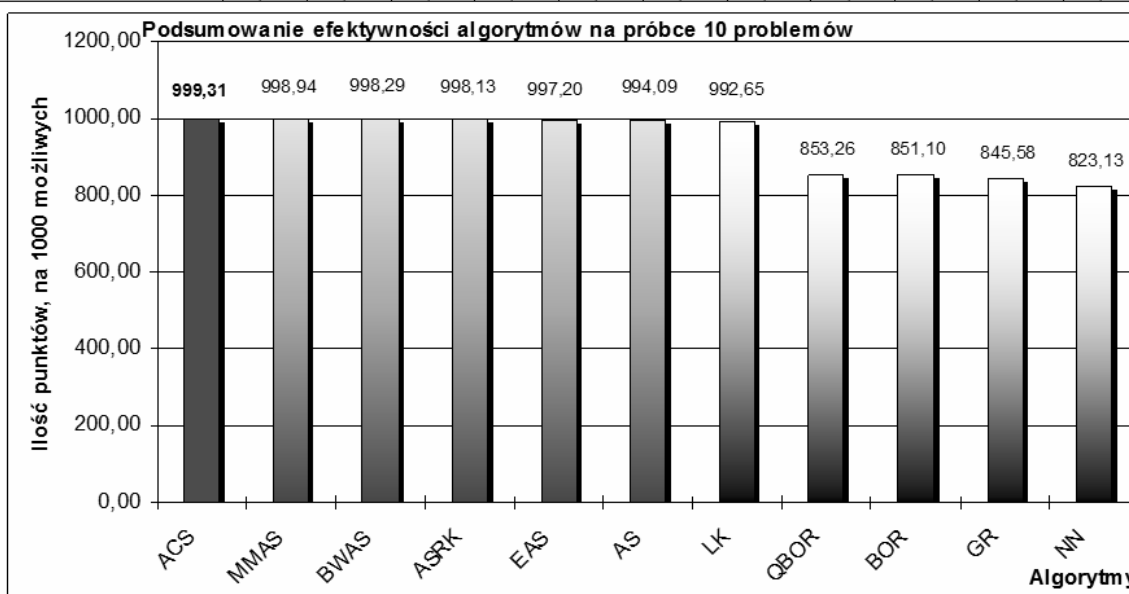
Zauważalną rzeczą jest skłonność wszystkich testowanych algorytmów do uzyskiwania gorszych rozwiązań wraz ze wzrostem wielkości problemu. Wyraźnym wskaźnikiem do tych skłonności był siódmy testowany problem pcb1173. Potwierdzeniem tego stwierdzenia był problem d1291 oraz każdy następny. Powyżej na wykresie podsumowującym wszystkie przeprowadzone testy możemy zobaczyć hierarchie wszystkich algorytmów oraz ilość otrzymanych punktów. Maksymalną ilością punktów jest 1000. Najbliżej tej wartości znajduje się przedstawiciel grupy ALM, czyli ACS z wynikiem 999,31 punktu. Co warto podkreślić pierwsze sześć miejsc pod względem efektywności poszukiwania optimum zajmują ALM. Kolejne lokaty to już przedstawiciele ALK i ich zdecydowany lider, czyli algorytm LK, który uznawany jest za najlepszy ALK do rozwiązania problemu TSP.

Marco Dorigo, który jest pomysłodawcą zastosowania zachowania mrówek do rozwiązywania problemów optymalizacyjnych, osiągnął ogromny sukces, którego potwierdzeniem jest powyższy artykuł. Algorytmy mrówkowe to nowa generacja algorytmów stosująca nowe podejście do bardzo trudnych problemów, podejście, które dało znakomite rezultaty. Algorytmy mrówkowe znajdują szereg innych zastosowań nie tylko do TSP, zastosowań które są wykorzystywane w sposób praktyczny w wielu dziedzinach życia. Celem tego artykułu było zastosowanie algorytmów mrówkowych do rozwiązań problemu TSP, który od samego początku istnienia algorytmów mrówkowych miał z nimi bardzo dużo do czynienia, ponieważ był to pierwszy problem, do którego zastosowano owe algorytmy, aby sprawdzić ich efektywność.

wysoko zorganizowanych. Dalszym kierunkiem jaki obrali sobie naukowcy jest wykorzystanie zdobytej wiedzy na temat koloni mrówek oraz przełożenie jej do zastosowań w świecie rzeczywistym. Określeniem, które opisuje nowoczesną mrówkę, która stosuje filozofię koloni mrówek w świecie rzeczywistym jest *s-bot*. Czym jest *s-bot*? Są to małe roboty skonstruowane w celu wykonywania wielu skomplikowanych zadań w świecie rzeczywistym wykorzystując pracę w grupie, lub inaczej mówiąc kolonii. Podstawą to stworzenia wspomnianych robotów jest *Swarm Theory*, która brzmi:

*A single ant or bee isn't smart, but their colonies are. The study of swarm intelligence [11] is providing insights that can help humans manage complex systems, from truck routing to military robots [12].*

Oprogramowanie		ACOTSP						Concorde TSP				
Plik *.tsp	optimum	ACS	MMAS	BWAS	Asrank	EAS	AS	Lin Kernighan	Quick Boruvka	Boruvka	Greedy	Nearest Neighbor
		ACS	MMAS	BWAS	ASRK	EAS	AS	LK	QBOR	BOR	GR	NN
	Ilość pkt na 1000	999,31	998,94	998,29	998,13	997,20	994,09	992,65	853,26	851,10	845,58	823,13
	średnia %	99,93%	99,89%	99,83%	99,81%	99,72%	99,41%	99,26%	85,33%	85,11%	84,56%	82,31%



Rys. 13 Podsumowanie efektywności algorytmów

## 6. KIERUNKÓW DALSZYCH BADAŃ

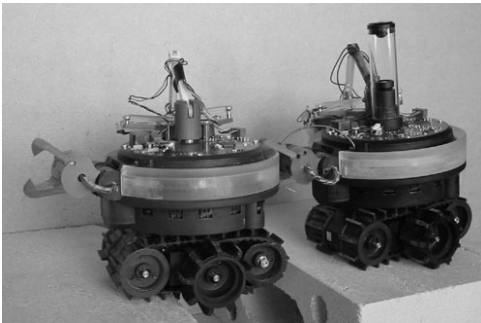
W tym punkcie zostaną przedstawione kierunki dalszych badań jakie zostały podjęte przez naukowców, wzorczujących się na koloniach mrówek jako stworzeń

Marco Dorigo był osobą koordynującą zakończony już projekt *Swarm-Bots: Swarms of self-assembling artefacts*, który był prowadzony w latach 2001-2005. Jego celem było studiowanie nowych podejść w stosunku do projektowania i wdrażania samo organizujących oraz samo-kontrolujących się robotów wynalezionych przez człowieka. Na samym początku

postawiono sobie szczegółowe cele do których zaliczamy:

- zaprojektowanie i utworzenie sprzętu w postaci: *s-bot*, *s-toy* oraz areny pna której będzie można przeprowadzać testy,
- zaprojektowanie i utworzenie symulatora,
- zaprojektowanie i utworzenie systemu kontroli *swarm-bot*'ów,
- integracja, testowanie oraz ocena rezultatu wszystkich powyższych czynności.

Załoženiami projektu było utworzenie od 30 do 35 identycznych *s-bot*ów które tworzyły by kolonię. Każdy z nich jest niezależną jednostką zdolną do wykonywania podstawowych zadań takich jak: nawigacja, umiejętność percepcji otaczającego go świata oraz możliwość zaczepienia się do innego obiektu.



Rys. 14 Kooperacja grupy robotów [7]

Pojedynczy *s-bot* jest w stanie komunikować się z pozostałymi jednostkami oraz ma możliwość połączenia się z nimi w sposób sztywny lub elastyczny. Połączenie kilku *s-bot*ów daje nam kolonię bot'ów (z ang. *swarm-bot*). Kolonia *s-bot*ów jest zdolna do wykonywania badań, nawigacji i transportu ciężkich przedmiotów na bardzo szorstkim terenie, szczególnie jeśli jeden *s-bot* ma poważne problemy w realizacji tego zadania w pojedynkę. *S-boty* komunikują się ze sobą za pomocą obręczy wyposażonej w diody świecące w różnych kolorach z możliwością mrugania o różnych częstotliwościach.

Kolejnym elementem, który był wykorzystywany w powyższym projekcie jest *s-toy*. Jest to specjalnie skonstruowany obiekt, który może być traktowany jako zdobycz lub oznaczenie gniazda dla *s-bot*ów. *S-toy* posiada wieżyczkę, która może być zdemonstrowana dzięki czemu jest możliwość obciążenia *s-toy*'a, maksymalnie do 2kg w celu zaimplementowania problemu współpracy w transporcie. Jest on wyposażony w identyczną obręcz jak *s-bot*'y co umożliwia zaczepienie *s-bot*'a do *s-toy*'a. Wieżyczka ma możliwość świecenia w dwóch kolorach. *S-toy* może dodatkowo emitować dźwięk w celach lokalizacyjnych.

Zostało podjętych wiele eksperymentów przy użyciu *s-bot*ów, które potrafiły wykazać się dużą samoorganizacją doprowadzając dane zadanie do realizacji. Do przykładowych eksperymentów można zaliczyć m.in.:

Scenariusz – jest eksperymentem polegającym na udaniu się przez *s-bot*'y do obiektu, którym w tym przypadku jest *s-toy* oraz przetransportowanie go do bazy. W tym miejscu można zobaczyć analogie do systemów mrówkowych. *S-toy*'em może być pokarm natomiast bazą, do której ma on zostać dostarczony jest gniazdo. Jest to typowy przykład organizacji w transporcie, gdzie pojedynczy osobnik nie daje rady w realizacji zadania i gdzie grupa *s-bot*ów musi się zorganizować w celu ukończenia zadania,

Przejsięcie przez dziurę – jest to przykład sytuacji, w której pojedynczy *s-bot* nie da sobie rady. Kolonia *s-bot*ów potrafi rozwiązać ten problem tak, aby kontynuować daną wędrówkę,

Współdziałanie w przemieszczaniu się – jest to przykład zadania, w którym *s-bot*'y przemieszczają *s-toy*'a i muszą zachować koordynację ruchów tak, aby sobie nawzajem nie przeszkadzać,

Patrząc z optymizmem na dotychczasowe dokonania naukowców i dynamizm, z jakim dziedzina *swarm-bot*ów się rozwija można wybiec trochę w przyszłość pokazując ogromne zastosowanie tych urządzeń, które mogą być przeznaczone do wykonywania wielu skomplikowanych zadań na ziemi, jak i w innych miejscach naszego układu słonecznego. Organizacja NASA, wyraziła duże zainteresowanie tego typu rozwiązaniem. Gdybyśmy wzięli pod uwagę sytuację, w której jeden z robotów podczas misji na Marsie nie mógłby kontynuować swojego zadania z różnych powodów, pozostałe *s-bot*'y mogłyby się same zorganizować w celu jego dokończenia. BBC news opublikowała na swojej stronie ciekawy artykuł pt. „*Smart future for swarm robots*”[13], w którym to umieszczony krótki filmik prezentujący możliwości organizacyjne *s-bot*ów. Nadaną im dwa zadania, gdzie jedno miało być wykonane przez 80% dostępnych *s-bot*ów a drugie przez pozostałe 20%. Przeprowadzono symulację w postaci zabierania kolejnych *s-bot*ów z areny, po której roboty same się organizowały doprowadzając do ustalonej na początku proporcji 80/20. Ciekawą perspektywą zastosować kolonii *s-bot*ów mogłoby być użycie ich do szukania ocalałych ludzi po jakiejś katastrofie oraz poszukiwanie innych zagrożeń, jakie mogłyby wystąpić.

## **Literatura**

1. MICHALEWICZ, Z., FOGEL, D.B., Jak to rozwiązać, czyli nowoczesna heurystyka, Warszawa, Wydawnictwo Naukowo-Techniczne Warszawa, 2006, s. 226,
2. BIGGS, N. L., LLOYD, E. K., WILSON, R. J., Graph Theory 1736-1936, [online], Oxford, Oxford University Press, 1998
3. HELSGANN, K., NGASSA, J-L., KIERKEGAARD, J., ACO and TSP, Roskilde University, may 2007
4. MILLER, P., Swarm Theory, [online], National Geographic Staff, lipiec 2007, <http://ngm.nationalgeographic.com/2007/07/swarms/miller-text>.
5. DORIGO, M., STÜTZLE, T., Ant Colony Optimization, London, The MIT Press Cambridge Massachusetts Institute of Technology, 2004, p. 69
6. DORIGO, M., GAMBARDELLA, M.I., Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. IEEE Transactions on Evolutionary Computation, 1997, p. 53-66,
7. PALMER, J., Smart future for swarm robots, Technology Reporter, BBC News, August 2008



UNIWERSYTET KAZIMIERZA WIELKIEGO

Wydział Matematyki, Fizyki i Techniki  
ul. Chodkiewicza 30

fax.: (+4852) 34-01-978  
tel.: (+4852) 34-19-264



[www.simis.ukw.eu.pl](http://www.simis.ukw.eu.pl)

# Studia i Materiały Informatyki Stosowanej

Czasopismo młodych pracowników naukowych, doktorantów i studentów

## Wprowadzenie

W dzisiejszej technice i naukach ścisłych trudno obejść się bez wspomaganie procesów badawczych i technologicznych metodami i narzędziami informatycznymi. Nowe czasopismo *Studia i Materiały Informatyki Stosowanej* skierowane jest do naukowców wykorzystujących szeroko rozumiane metody informatyczne w swoich pracach badawczych na polu różnych dyscyplin technicznych. Pozwoli ona skonsolidować różne w swoich obszarach badawczych dyscypliny techniczne z punktu widzenia podobnych metod komputerowych wspomagających te dziedziny. Do aktywnego udziału zaproszeni są wszyscy pracownicy, doktoranci i studenci, szczególnie ci reprezentujący szeroko rozumianą technikę, którym bliska naukowo jest dziedzina Informatyki Stosowanej. Szczególne zaproszenie kierujemy do młodych pracowników naukowych, do osób prowadzących koła naukowe. Naszym pragnieniem jest aby SiMIS stało się forum gdzie doktoranci, czy studenci-dyplomanci opublikują swoje pierwsze publikacje, często z promotorem, i wkroczą na drogę działalności naukowej.

## Zakres tematyczny

- Różne dziedziny Informatyki (*Sztuczna Inteligencja, Kryptologia, Inżynieria Oprogramowania, Algorytmika, Programowanie Równoległe, Sieci komputerowe, itp.*),
- Komputerowe wspomaganie projektowania CAx,
- Symulowanie komputerowe procesów w technice,
- Symulacje i modelowanie w naukach ścisłych,
- Przetwarzanie danych pomiarowych,
- Mechatronika,
- Wszelkie zastosowania technik komputerowych w nauce i technice.

## Ważne daty

- 30.11.2009** Ostateczny termin nadsyłania artykułów
- 20.12.2009** Ostateczna akceptacja artykułów i zakończenie recenzji
- 10.01.2010** Rozpoczęcie procesu wydawniczego przez Wydawnictwo



UNIwersytet KAZIMIERZA WIELKIEGO

Wydział Matematyki, Fizyki i Techniki  
ul. Chodkiewicza 30

fax.: (+4852) 34-01-978  
tel.: (+4852) 34-19-264



www.simis.ukw.eu.pl

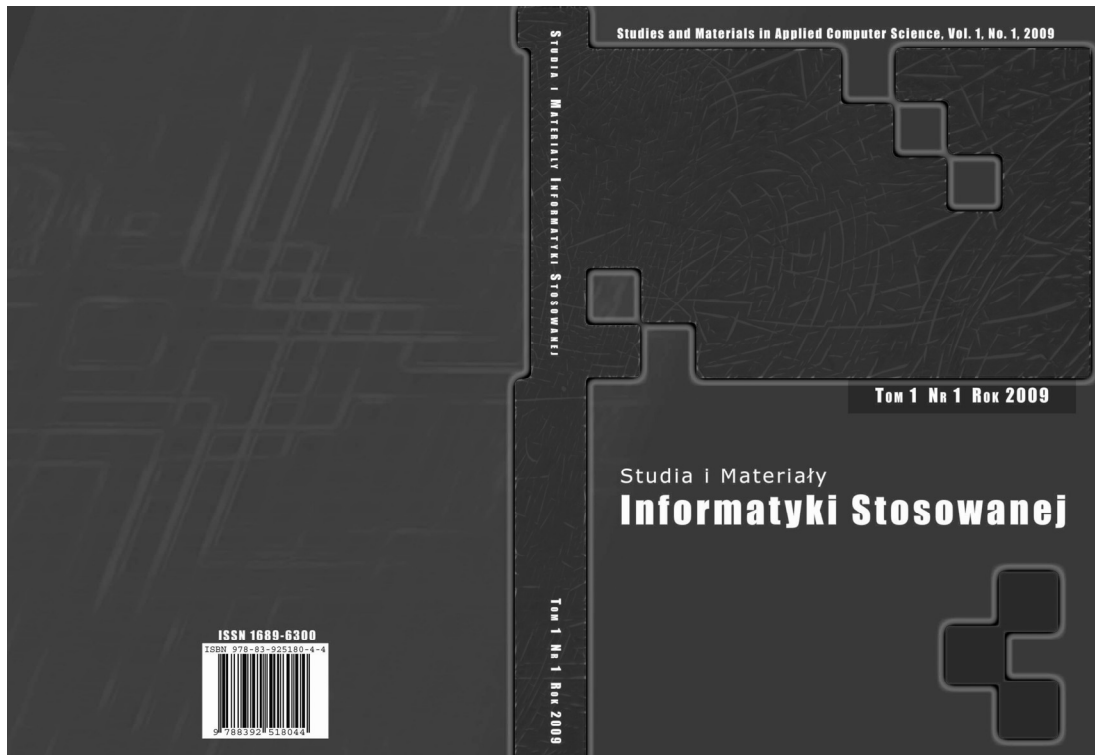
## Dla autorów

Studia i Materiały Informatyki Stosowanej są czasopismem ukazującym się co pół roku. Możliwe jest wyłącznie nadsyłanie artykułów, których wcześniej nie publikowano, wszystkie artykuły są recenzowane.

- Studia i Materiały Informatyki Stosowanej są wydawnictwem recenzowanym
- Studia i Materiały Informatyki Stosowanej są publikacją zgłoszoną do planu wydawniczego 2008/2009 Wydawnictwa UKW
- Autorem może być nauczyciel akademicki, doktorant lub student oraz każda inna osoba zainteresowana tematyką.
- Format artykułu w załączniku, grafika będzie konwertowana do skali szarości
- Język publikacji: polski z obowiązkowym streszczeniem w j.angielskim, lub całość artykułu w j.angielskim.
- Zgłoszenie artykułu następuje przez wysłanie maila z wypełnionym plikiem „SiMIS2009 zgłoszenie” na adres: jczerniak@ukw.edu.pl
- Artykuły napisane z użyciem szablonu „SiMIS2009 szablon” prosimy przesłać na adres: jczerniak@ukw.edu.pl

## Kontakt

- dr inż. Jacek Czerniak, kontakt: tel. (052) 341 92 64, p.216A, jczerniak@ukw.edu.pl
- dr inż. Marek Macko, kontakt: tel. (052) 341 93 30, p.04, mackomar@ukw.edu.pl





Faculty of Mathematics, Physics  
and Technical Science  
ul. Chodkiewicza 30

KAZIMIERZ WIELKI UNIVERSITY IN BYDGOSZCZ

fax.: (+4852) 34-01-978  
tel.: (+4852) 34-19-264



[www.smacs.ukw.edu.pl](http://www.smacs.ukw.edu.pl)

2009-10-09

# Studies and Materials in Applied Computer Science

Journal of young researchers, PhD students and students

## Introduction

Modern engineering and science cannot do without computer-aided research and technologic processes. The new journal, Studies and Materials in Applied Computer Science, is aimed at scientists who, in their research work on diverse engineering branches, use computer science methods in their broad sense. With that journal, it will be possible to consolidate different engineering branches that are aided by common computer-related methods. We would like all employees, PhD students, students and specially those who represent engineering in its broad sense and whose research works are closely related to Applied Computer Science to feel invited to actively cooperate with the journal. We address that invitation especially to young researches and to leaders of scientific circles. We would like SMACS to become a forum where PhD students or MSc. candidates publish their first papers, often together with their thesis supervisor, and thus they enter the world of scientific activity.

The editorial staff is going to publish original papers of experts of world renown as well as young scientists both in Polish and in English language.

An important part of our publishing programme will be special editions prepared by world-renown authors.

We also plan to publish review papers presenting the state of knowledge in a given branch of science as well as special papers written by most outstanding Polish and foreign scientist, who present original, sometimes controversial attitude towards challenges that are put out to computer science in its broad sense by the modern world. Those specially ordered papers will be published in a series called "Meetings with Masters". We particularly want young computer science specialists who enter the world of science or begin their professional career to learn opinion of well-known experts of their branch. Hence the "master-student" relationship that has been present in science and in culture since the dawn of time will be continued by our journal.

## Range of topics

- Different Computer Science branches (*Artificial intelligence, Cryptology, Software Engineering, Theoretical Computer Science, Concurrent programming, Computer networks, etc.*),
- Computer Aided Design CAx,
- Computer Aided technological process simulation,
- Simulation and modelling in science,



- Measurement data processing,
- Mechatronics,
- Any other computer technology applications in science and engineering.

## Important deadlines

**30.11.2009** Final deadline to receive papers

**20.12.2009** Final acceptance of papers and completion of review process

**10.01.2010** Beginning of the publishing process by publishing company

## To authors

Studies and Materials in Computer Science is a journal that comes out twice a year. Papers sent to us by their authors cannot be published before. All the articles will be subject to review. By now, research studies were published under the name Applied Computer Science Symposium, a compact publication in monograph form and it included papers presented during meetings and symposiums.

- Studies and Materials in Applied Computer Science is a publication subject to review
- Studies and Materials in Applied Computer Science is a journal entered into the publishing plan of UKW for the year 2008/2009.
- The author of a paper can be an employee, a PhD student, a student or any other person interested in this subject.
- See the attachment with the format of the paper. Be aware that graphics will be converted into greyscale.
- Language of a publication: Polish plus mandatory summary in English or an entire paper in English.
- Papers should be submitted by sending an e-mail including completed file „SMACS2009\_request” to the address: [jczerniak@ukw.edu.pl](mailto:jczerniak@ukw.edu.pl)
- Papers written using the „SMACS2009\_template” template should be sent to the address: [jczerniak@ukw.edu.pl](mailto:jczerniak@ukw.edu.pl)

## Contact:

- PhD. eng. Jacek Czerniak, contact: Phone: (+4852)341-92-64, room 216A, [jczerniak@ukw.edu.pl](mailto:jczerniak@ukw.edu.pl)
- PhD. eng. Marek Macko, contact: Phone: (+4852)341-93-30, room 04, [mackomar@ukw.edu.pl](mailto:mackomar@ukw.edu.pl)

**ROBOTYKA DLA  
PRZYSZŁOŚCI**

**WWW.ROBOSHOP.PL  
SKLEP DLA ROBOTYKÓW**



**TYLKO U NAS  
ZNAJDZIESZ NAJNOWSZE  
PRODUKTY ZE ŚWIATA ROBOTYKI**

**JESTEŚMY W STANIE WYPOSAŻYĆ  
KAŻDE LABORATORIUM ROBOTYKI**

**SZCZEGÓLNIIE POLECAMY ZESTAWY EDUKACYJNE:**

- **BIOLOID**
- **LEGO MINDSTORMS NXT**



**ZAPRASZAMY DO WSPÓŁRACY!!!**

**RoboSHOP.pl**

ul. Trzy lipy 3 | 80-172 Gdańsk  
tel./fax (058) 739 61 61 | biuro@roboshop.pl  
**www.roboshop.pl**

**ROBO  
SHOP.PL**

**ROBOTYKA DLA  
PRZYSZŁOŚCI**

**WWW.ROBOSHOP.PL  
SKLEP DLA ROBOTYKÓW**



**TYLKO U NAS  
ZNAJDZIESZ NAJNOWSZE  
PRODUKTY ZE ŚWIATA ROBOTYKI**

**JESTEŚMY W STANIE WYPOSAŻYĆ  
KAŻDE LABORATORIUM ROBOTYKI**

**SZCZEGÓLNIIE POLECAMY ZESTAWY EDUKACYJNE:**

- BIOLOID
- LEGO MINDSTORMS NXT



**ZAPRASZAMY DO WSPÓŁRACY!!!**

**RoboSHOP.pl**

ul. Trzy lipy 3 | 80-172 Gdańsk  
tel./fax (058) 739 61 61 | biuro@roboshop.pl  
**www.roboshop.pl**

**ROBO  
SHOP.PL**