Proceedings of the 2nd INTERNATIONAL CONFERENCE ON APPLICATIONS IN INFORMATION TECHNOLOGY (ICAIT-2016)

October 6 – 8, 2016
University of Aizu
Aizu-Wakamatsu
Japan

The University of Aizu Press
2016

Proceedings of the

2nd International Conference on Applications in Information Technology

Aizu-Wakamatsu, Japan October 6 – 8, 2016

Edited by

Evgeny Pyshkin

Vitaly Klyuev

Alexander Vazhenin

Hosted by

The University of Aizu

In Cooperation with

St. Petersburg State University

Peter the Great St. Petersburg Polytechnic University

Novosibirsk State University

The University of Aizu Press
2016

会津大学

ISBN 978-4-900721-04-3

Additional copies may be ordered from:
The University of Aizu Press
University of Aizu
Aizu-Wakamatsu, Fukushima-ken 965-8580
Japan

2nd INTERNATIONAL CONFERENCE ON APPLICATIONS IN INFORMATION TECHNOLOGY

October 6 – 8, 2016

Table of Contents

ORGANIZING AND PROGRAM COMMITTEES

HONORARY CHAIRS	Ryuichi Oka	*President,* *University of Aizu, Japan*
	Leon Petrosjan	*Dean,* *Faculty of Applied mathematics – Control Processes,* *St. Petersburg State University, Russia*
	Vladimir Zaborovsky	*Director,* *Institute of Computer Science and Technology,* *Peter the Great St. Petersburg Polytechnic University, Russia*
	Mikhail M. Lavrentiev	*Dean,* *Faculty of Information Technologies,* *Novosibirsk State University, Russia*
CONFERENCE CHAIRS	Vitaly Klyuev	*University of Aizu, Japan*
	Alexander Vazhenin	*University of Aizu, Japan*
PROGRAM COMMITTEE CHAIR	Evgeny Pyshkin	*University of Aizu, Japan*
PROGRAM COMMITTEE	Thomas Baar	*Hochschule für Technik und Wirtschaft Berlin, Germany*
	Natalia Bogach	*Peter the Great St. Petersburg Polytechnic University, Russia*
	Paolo Bottoni	*University of Rome, Italy*
	John Brine	*University of Aizu, Japan*

PROGRAM COMMITTEE (CONT.)

Alfredo Capozucca	*University of Luxembourg, Luxembourg*
Hapugahage Thilak Chaminda	*Informatics Institute of Technology, Sri Lanka*
Ruth Patricia Cortez	*Similatio, Japan*
Vlatko Davidovski	*Cognizant Business Consulting, Switzerland*
Vladimir Dobrynin	*St. Petersburg State University, Russia*
Mikhail Glukhikh	*JetBrains, Peter the Great St. Petersburg Polytechnic University, Russia*
Nicolas Guelfi	*University of Luxembourg, Luxembourg*
Mohamed Hamada	*University of Aizu, Japan*
Yannis Haralambous	*Institut Mines-Télécom, Télécom Bretagne, France*
Hucine Hassan	*Universitat Politècnica de València, Spain*
Vladimir Itsykson	*Peter the Great St. Petersburg Polytechnic University, Russia*
Qun Jin	*Waseda University, Japan*
Vitaly Klyuev	*University of Aizu, Japan*
Sergei Krutolevich	*Belarusian Russian University, Belarus*
Andrey Kuznetsov	*Motorola Solutions Inc., Peter the Great St. Petersburg Polytechnic University, Russia*

PROGRAM COMMITTEE (CONT.)

Petri Laitinen	*Karelia University of Applied Sciences, Finland*
Rüdiger Lunde	*Ulm University of Applied Sciences, Germany*
Viacheslav Marakhovsky	*Peter the Great St. Petersburg Polytechnic University, Russia*
Calkin Suero Montero	*University of Eastern Finland, Finland*
Maxim Mozgovoy	*University of Aizu, Japan*
Kendall Nygard	*North Dakota State University, USA*
Kohei Ohno	*Meiji University, Japan*
Mikhail Okrepilov	*Peter the Great St. Petersburg Polytechnic University, Russia*
Vladimir Oleshchuk	*University of Adger, Norway*
Benoît Ries	*University of Luxembourg, Luxembourg*
Alexey Romanenko	*Novosibirsk State University, Russia*
Hiroshi Saito	*University of Aizu, Japan*
Roman Shtykh	*CyberAgent Inc, Japan*
Nikolay Smirnov	*St. Petersburg State University, Russia*
Kari Smolander	*Lappeenranta University of Technology, Finland*
Alexander Vazhenin	*University of Aizu, Japan*
Senzhang Wang	*Nanjing University of Aeronautics and Astronautics, China*

PROGRAM COMMITTEE (CONT.)	Ying-Hong Wang	*Tamkang University, Taiwan, China*
	Yang Zhibin	*Nanjing University of Aeronautics and Astronautics, China*
PROCEEDINGS EDITORS	Evgeny Pyshkin	*University of Aizu, Japan*
	Vitaly Klyuev	*University of Aizu, Japan*
	Alexander Vazhenin	*University of Aizu, Japan*
EDITOR-IN-CHIEF	Yasuhiro Abe	*The University of Aizu Press, University of Aizu, Japan*
PROCEEDINGS COVER DESIGNER	Evgeny Pyshkin	*University of Aizu, Japan*
LOCAL ARRANGEMENTS TEAM	Yuriko Nagashima	*University of Aizu, Japan*
	Aya Anazawa	*University of Aizu, Japan*

GREETING FROM THE PRESIDENT OF THE UNIVERSITY OF AIZU

It is my honor to welcome all participants to the 2nd International Conference on Applications in Information Technology.

We have organized this annual event within the framework of MEXT's Top Global University Program initiated in 2014. We are working in close cooperation with our Russian partner universities: Saint-Petersburg State University, Peter the Great Saint-Petersburg Polytechnic University, and Novosibirsk State University.

The best undergraduate, master, and PhD students of the aforementioned universities are meeting here to discuss their latest scientific achievements. At this conference, the participants come to understand the spirit of scientific events, establish their first scientific contacts with colleagues, and exchange views on scientific problems. This is an important tradition of this event. Most of the student papers are co-authored with their supervisors. This is also a good tradition of this conference, and it helps us to select high quality submissions for presentation.

This year, we have made further progress: Two reputable journals have agreed to publish special issues that will contain extended versions of selected papers presented at this conference. One journal is *Informatica*; the other is *Information*. These special issues are a recognition of the high scientific level of the publications in the proceedings from previous years; this is truly an outstanding achievement of our conference.

I also would like to mention another good tradition of this conference: At the end of the twentieth century, we pioneered in running conference sessions over the Internet connecting remote sites with the main event. We are using videoconferencing this year to expand the conference room for the presenters located in other countries.

I would like to thank our international team of reviewers. They did a wonderful job: Their suggestions and advice were very helpful for the participants of the accepted papers in making the necessary revisions and thereby elevating the quality of publications.

Finally, I would like to extend my thanks to all of the participants for their efforts in maintaining the high scientific level of the 2nd International Conference on Applications in Information Technology.

Ryuichi Oka

***Ryuichi Oka**, Ph.D.*

President of the University of Aizu



WELCOME FROM THE CHAIRS

The 2nd International Conference on Applications in Information Technology (ICAIT-2016) is an excellent example of the ongoing collaboration among the international participants and organizers. An important component of education at any university is the involvement of students in international scientific research. Development of the interdisciplinary research and academic projects is an inherent part of our collaboration programs.

We are happy that the conference brings together the scientists advancing the design, development, use, and evaluation of information technology applications from many respected research institutions all around the world.

ICAIT-2016 accumulates good traditions established in the past conferences organized in the University of Aizu including *The Conference on Humans and Computers* in 1998-2010, its successor, *The 2012 Joint International Conference on Human-Centered Computer Environments* and *The 2015 International Workshop on Applications in Information Technology*. This year we received 62 submissions, so the competition was really hard. All submissions were peer-reviewed in a single-blind review process. A total of 34 papers were selected for presentation at the conference. We believe that the conference meetings and discussions will anticipate the future growth of the joint research activities among our partners.

We thank our partner journals *Informatica*, *Information* and *St. Petersburg State Polytechnic University Journal* for their support for the distinguished papers presented at the conference, which, after further extensions, will be recommended for submission and publication in the above mentioned journals.

We would like to thank our outstanding team of volunteer educators and support staff who have worked very hard to put together ICAIT-2016. We also acknowledge the invaluable assistance of the international referees who agreed to join the conference program committee.

We hope you enjoy the 2016 edition of the ICAIT!

Evgeny Pyshkin, Vitaly Klyuev *and* ***Alexander Vazhenin***

CONFERENCE INFORMATION

MISSION STATEMENT

The main objective of the 2nd Conference on Applications in Information Technology is to foster rich creativity in students' research works and to encourage students and young scientists to participate actively in open discussions with their colleagues. This conference is a place for the scientific presentations of young researchers representing not only the universities organizing this event but also many partner universities from all over the world.

2016 CALL FOR PAPERS SUMMARY

The 2nd Conference on Applications in Information Technology invited papers presenting new advances and research results in various fields of information technology applications. ICAIT-2016 is a multi-disciplinary conference bringing together under-graduate, graduate, master and Ph.D. students, academic scientists, researchers and scholars working in the domains of IT development and applications. In 2016 we particularly welcomed high quality contributions in the areas which include but not limited to:

- 3D Modeling, Virtualization and Visualization
- Cloud Computing and Networking
- Computer Assisted Learning
- Computer Systems and Architectures
- Control Systems
- Game Development
- Human-Centric Computing
- Information Retrieval and Data Science
- Intelligent and Embedded Systems
- Knowledge Engineering
- Machine Learning
- Machine Vision and Pattern Recognition
- Mobile and Cross Platform Applications
- Modeling and Simulation
- Software Design and Engineering
- Web Engineering

WORKSHOP INFORMATION SPONSOR

St. Petersburg State Polytechnic University Journal.

Computer Science, Telecommunication and Control Systems

http://ntv.spbstu.ru/telecom/

CONFERENCE PARTNER JOURNALS

Informatica 1977

01010 01010 01010 information

Trends and Opportunities of Training Highly Qualified IT Specialists in Accordance with International Standards

Sergei S. Voitenko
St. Petersburg State University
62 Tchaikovsky st.
191123 St. Petersburg Russia
s.voitenko@spbu.ru

Lyudmila F. Vyunenko
St. Petersburg State University
62 Tchaikovsky st.
191123 St. Petersburg Russia
l.vyunenko@spbu.ru

Alexander V. Yurkov
St. Petersburg State University
62 Tchaikovsky st.
191123 St. Petersburg Russia
a.v.yurkov@spbu.ru

ABSTRACT
The aim of the research is to analyze the conformity of training curriculum in the area of business informatics at the Saint Petersburg State University (SPBSU) to the requirements of the SWEBOK standard. The results of the research would be setting the conditions necessary for training Russian PhD-students according to the international standards of Software Engineering (SE) and recommendations for the corresponding improving of IT-education programs in the University. International cooperation in the field of doctoral training is supported by the project Joint Programs and Framework for Doctoral Education in Software Engineering in the European Union Erasmus+ Program Capacity Building in Higher Education. The overall aim of the project is to support the development, modernization and internalization of SE higher education in the target countries according to the SWEBOK international standard. The standard includes the following areas: Research Methods in the age of Software as a Service, Advanced methods and tools of SE, Problem-based learning in the field of SE, Human-computer interaction in SE, Models of SE and modeling, Mathematical and computational foundations of SE, SE from the economics and business point of view. The last area of the research is widely represented at the Faculty of Economics of Saint Petersburg State University. Business informatics courses are taught here by highly qualified staff for several years.

Categories and Subject Descriptors
K.3.2 [**Computers and Education**]: Computer and Information Science Education - *IT education standards, international cooperation*

General Terms
Management, Economics, Standardization

Keywords
Business informatics, IT education standards, Software Engineering Body of Knowledge (SWEBOK), International cooperation, PhD training

1. INTRODUCTION
The current graduate education system in many countries is based on an apprenticeship model, wherein lab heads train younger researchers in the craft of research. This system has been prominent since the 1800s, when the first 'modern' PhD was awarded by the University of Berlin. Although the scientific enterprise has changed dramatically since then, the PhD system has not. Modernizing the PhD could improve training in areas of research ranging from reproducibility to experimental design and entrepreneurship. It could also help to solve the bottleneck problem by equipping doctorate holders with soft skills that make them more employable wherever they go [4].

The last few decades have witnessed an assortment of rapid and transformative changes in the needs of society and in the institutions created to respond to these needs. As the economy shifts from based on the production of goods to based on the production of knowledge and information it creates a demand for new types of learners and innovators in every profession. Employers and administrators in every sector now report that they expect doctoral programs to create well-rounded disciplinary experts who have the ability to be leaders in their fields and are capable of creating real-world value from knowledge and discovery. In spite of these economic realities, universities—and doctoral programs specifically — largely continue to view Ph.D. training as a means of reproducing the status quo [12].

In Russia for more than 25 years the reform of higher education is being implemented in accordance with the Bologna system. The two main objectives of the reforms are introduction of the new federal state educational standard and transition to the three cycle bachelor-master-doctorate system. The last stage of this multi-level higher education model is particularly important as it contributes to the modern world science. As we see it, the major problem of this progressive essential transformations is that the integrated training conveyor for highly qualified scientific personnel has not been still fully implemented [8]. The comparison of the above mentioned three-cycle process

and the traditional stepped educational procedure of the Russian higher school shows that a specialist in Russia receives more high level of education and training than a bachelor in Bologna model. Graduates from leading Russian universities qualified as specialists in terms of final theses are comparable with foreign masters. However, modern domestic postgraduate training, especially for applied areas, is significantly less efficient than the Institute of PhD-doctoral training. The transfer and adaptation of this institute in the Russian education system is still not implemented. For example, in Saint Petersburg State University there are only a few examples of PhD defenses. Thus, the transition from a stepped model of higher education, which differed Russian higher school, to a two-level model should be supplemented with a well-developed doctoral training, carried out in accordance with international standards. In the field of Software Engineering, one of the most recognized standards is the Software Engineering Body of Knowledge – SWEBOK [1]. For bachelor degree program "Business Informatics" and master degree program "Informational business Analytics" presented at the Faculty of Economics of Saint Petersburg State University another important set of recommendations is standard Business Analysis Body of Knowledge – BABOK [5]. Training at these programs is conducted in the following areas: Investment Analysis of SE Projects, Risk Analysis and Risk Management of SE Projects, Business Intelligence for SE Costing, Technics and Methodology of Simulation Modeling in SE, Perfect Software Engineer Competences, Economics and Competitiveness of SE Education.

2. SPBSU OPPORTUNITIES and TRADITIONS in the DEVELOPMENT of its OWN EDUCATIONAL STANDARDS

In the field of higher education in the Russian Federation three generations of state educational standards have been developed and put into operation over the past 25 years, mandatory for compliance for the universities competing for state accreditation. The main trend of their evolution has been the expansion of academic freedom of universities in shaping the structure and content of education. Federal state educational standards (FSES) were put in place in 2011. FSES advantage was the use of credits compatible with the ECTS to assess the complexity of the mastering of educational programs. In 2012, a new edition of these standards FSES 3+ was developed. In this standard the emphasis is put on the competence of the graduates of educational programs and expanding the rights of educational institutions in determining the structure and content of the programs. Currently FSES 3+ format developed 691 standards within 57 enlarged groups of directions and specialties of undergraduate specialties, graduate and postgraduate studies. A special feature was FSES 3+ division undergraduate and graduate educational programs in academic and applied, depending on the type of the main professional activity of graduates. At the freedom of universities in the planning of the professional competencies of graduates realized in the level of training of highly qualified personnel.

Only a few of the leading Russian universities (including the Saint Petersburg State University) today, which by the Federal Law "On Education in the Russian Federation" are allowed to develop and implement their own educational standards have the right to develop their own sets of educational standards, determine the structure and content of educational programs at their own direction at all levels of higher education. Saint Petersburg State University's own educational standard approved in 2014 [7], is set up for each direction of training, corresponding to GEF, bachelor, specialty, and master. Master includes several models: academic, academically-oriented and practice-oriented. According to the educational standards of the Saint Petersburg State University, each graduate should possess general competence in terms of: improvement and development of their general intellectual and cultural level; use knowledge of modern achievements of science and education in solving the educational and professional goals; the independent development of new methods of research, changing the profile of their professional activities; independent acquisition of information technologies by means of new knowledge and skills and use them in practice, including in new areas; work with professional texts in English and Russian languages, and others.

The standard specifies requirements for the results of the mastering of educational programs depending on peculiarities directions of training, also includes a list of formed professional competencies. The basis is to match them to the requirements of employers and the results of the mastering of the appropriate level of educational programs in leading universities of the world. For each direction of training the educational standard of the Saint Petersburg State University establishes a list of required professional foreign language competences, which must possess a graduate of magistracy. The standard defines the requirements for the structure of educational programs, the conditions of their implementation, and the quality assessment at the level of the leading universities in the world. Appendix to the educational standard is a list of master's degree programs in directions of preparation, corresponding FSES. It should be noted that the educational standard at the level of training of highly qualified personnel – graduate and PhD-doctorate – in the Saint Petersburg State University is currently missing.

In a recent paper [3] states that elaboration by universities their own educational standards in directions not listed in the Enumeration of Federal state educational standards is a rare phenomenon. However, in our view, this practice should be developed in the near future. The impetus for the development of educational standards will be a need for interdisciplinary programs, mastering which will organically combine the advanced knowledge, skills and professional experience in various fields of fundamental and applied researches of natural science, technical and humanities. The logic of the formation of knowledge-based economy necessitates the development of educational standards of the country's leading universities. Own educational standards of universities contribute to the realization of their missions, taking into account the traditions and development strategies. Standards of leading universities are working to increase their competitiveness on the national and international levels, affect the positioning of universities in the world scientific and educational space. The main direction of development of educational standards independently established by leading Russian universities, is, obviously, to create conditions for the development and implementation of

world-class interdisciplinary educational programs needed for training, sought after a dynamic science and high technology manufacturing. All this is a positive impact on improving the quality of the Russian higher education system.

3. The DEVELOPMENT of the IT EDUCATIONAL DIRECTION at SPBSU

The leading Russian universities – Moscow State University and Saint Petersburg State University – since the mid 80-ies traditionally taught IT at a high theoretical level together with serious mathematical training. These training programs improved in accordance with international educational standards. Immediately after the appearance of «Computing Curricula 2001: Computer Science» and «Software Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering» Saint Petersburg State University published their translations into Russian. In 2003, the Moscow State University proposed and adopted a new educational direction "Information Technology", as well as educational standards of Bachelor and Master of IT, focused on the training of professionals demanded in research projects, industry and business [9]. Those were the first Russian educational standards developed with the advice of "Computing Curricula", but unlike the foreign educational programs they have preserved the fundamental nature of training, in particular, a more in-depth and focused mathematical preparation. Already from the 2003/04 school year, the Moscow State University and the Saint Petersburg State University students have begun to conduct training on this educational direction, and in 2006 at Saint Petersburg State University was carried out serious work on bringing the curriculum "Information Technology" in compliance with the recommendations «Computing Curricula: Software Engineering» [10]. The next step was the opening in 2011 in Saint Petersburg State University the direction of training bachelors and masters on the "Software Engineering" speciality. The draft program was focused on the balance between fundamental (mathematical methods, algorithms, mathematical logic), applied technological (private algorithms, technologies and paradigms) and humanitarian (project management, economic and legal basis of the production of software, the theory of inventive problem solving) components.

4. INTERNATIONAL COOPERATION in SE DOCTORAL EDUCATION

Due to the fact that the educational standards for the training of graduate students and PhD-students for IT and SE directions of the Russian legislation are not provided, and the ability to assign their own academic degrees, the Saint Petersburg State University is making efforts for setting the conditions necessary for training Russian PhD-students according to the international standards of SE and recommendations for the corresponding improving of IT-education programs in the University.

In particular, international cooperation in the field of higher qualification scientific personnel training is supported by the project Joint Programs and Framework for Doctoral Education in Software Engineering in the European Union Erasmus+ Program Capacity Building in Higher Education. One global aim of capacity building would be to change the composition of staff at Higher Education Institutions to make this more representative. Secondly, structural problems within the higher education sector result in inequalities between institutions in terms of staff qualifications, staff/student ratios, availability of postgraduate programs, availability of research funding, library and information services, information technology, management capacity, and infra- structural support [2]. The overall aim of the mentioned project is to support the development, modernization and internalization of SE higher education in the target countries according to the SWEBOK international standard and at the same time to help all partner countries to strengthen their PhD programs and make them more attractive for international students. The project will facilitate building and offering of sustainable multi-disciplinary and multi-regional perspective programs for educating the next generation of software engineering professors [11].

The project proposes training for qualified teaching staff of all partner universities and for PhD students in the target countries. During the 2015-2018 eleven European universities, including the Saint Petersburg State University, will share their experience in creating educational environment according to the international standards. The program is structured in several intense schools which will be organized by different universities. It will help to share specific expertise of partner universities in different areas. Each university can nominate PhD students to attend the different schools. Target countries also can train their staff in these schools. These schools are held in different knowledge areas and disciplines providing typical PhD student information which he or she cannot acquire in one specific university. Individual PhD students can select the best suited schools according to their research profile and create personal path in the proposed program. PhD students and their supervisors will be offered courses on advanced techniques of SE and reviews of the best practices of research projects. Help will be provided in preparation of high-quality scientific publications on the subject, which includes the following areas: Research Methods in the age of Software as a Service, Advanced methods and tools of SE, Problem-based learning in the field of SE, Human-computer interaction in SE, Models of SE and modeling, Mathematical and computational foundations of SE, SE: A View from the point of view of economics and business. The last area of the research is widely represented in the Faculty of Economics of Saint Petersburg State University, business informatics courses being taught here for several years by highly qualified staff [6]. Training is conducted in the following areas: Investment Analysis of SE Projects, Risk Analysis and Risk Management of SE Projects, Business Intelligence for SE Costing, Technics and Methodology of Simulation Modeling in SE, Perfect Software Engineer Competences, Economics and Competitiveness of SE Education. Of course, we understand that SWEBOK standards have the rather limited intersections with educational programs in the direction of business informatics, and take into account the presence of BABOK and other IT standards. However, the problem of the development of educational standards for the preparation of highly qualified scientific personnel is so acute for the university that we use all the opportunities.

5. CONCLUSION

With the nature of jobs in the current market for SE and BI graduates permanently transformed by a changing global economy, graduate education now finds itself with an opportunity to reinvent itself as well. Contemporary Ph.D. students are beginning to realize that their training must change and are demanding that universities provide graduates with the skills necessary to operate professionally within the new economy. Recent research on graduate students [12] emphasize that while the technical and disciplinary-specific training received during doctoral programs is useful, they are missing basic professional skills that are essential in the contemporary economy. The findings of this study suggest that many students in doctoral programs are not only aware that they are lacking these skills, they are willing and looking for flexible, interesting, efficient, and relevant ways to develop them over the course of their graduate programs.

6. ACKNOWLEDGMENTS

The research is partially supported by the Russian Foundation for Basic Research (RFBR), grant 16-06-00221.

7. REFERENCES

[1] Bourque, P. et al. *Guide to the Software Engineering Body of Knowledge, Version 3.0* - IEEE Computer Society, 2014. Available at http://www.swebok.org (accessed 21.06.2016).

[2] CHET 2002. *Capacity Building Initiative in Higher Education* - Centre for Higher Education Transformation, 2002. Available at http://www.chet.org.za/download/file/fid/198 (accessed 07.07.2016).

[3] Chuchalin, A. *Educational Standards of the Leading Russian Universities.* - Higher Education in Russia, 2015, no 4, pp. 14 25.

[4] Gould, J. *How to Build a Better PhD.* - Nature, Dec 2015, vol. 528, pp. 22 25.

[5] IIBA 2016. *A Guide to the Business Analysis Body of Knowledge (BABOK Guide), Version 3.0* - International Institute of Business Analysis, 2016. Available at http://www.iiba.org/babok-guide.aspx (accessed 07.07.2016).

[6] Khalin, V., Yurkov, A. et al. *Prikladnaya informatika v ekonomike. Sbornik programm uchebnykh distsiplin po spetsial'nosti 080801 (351400).* [Applied Informatics for Economy. Collection of educational programs for specialty 080801 (351400)] – Saint Petersburg, 2008, 260 p. (in Russian).

[7] SPbGU 2014. *Obrazovatel'nye standarty* [SPbSU. Educational standards]. – Saint Petersburg State University, 2014. Available at http://spbu.ru/structure/documents/mm19xm7g.html (in Russian, accessed 07.07.2016).

[8] Sukhomlin V.A. *On the outcome of the higher school reform (reflections of Russian professors)* – Journal of Applied Informatics, 2011, vol. 32, no. 2, pp. 41–48 (in Russian). Available at http://www.appliedinformatics.ru/general/upload/articles_preview/pi3210-41_fs.pdf (accessed 07.07.2016).

[9] Sukhomlin, V.A., Sukhomlin, V.V. *Kontseptsiya novogo obrazovatel'nogo napravleniya* [The concept of a new educational direction] *Otkrytye sistemy* - Open Systems Journal, 2003, no. 2, pp. 31-34 (in Russian).

[10] Terekhov, A.A., Terekhov, A.N. *Computing Curricula: Software Engineering i rossiiskoe obrazovanie* [Computing Curricula: Software Engineering and Russian Education] *Otkrytye sistemy* - Open Systems Journal, 2006, no. 8, pp. 31-34 (in Russian).

[11] Voitenko, S., Vyunenko, L., Yurkov, A. *Business Informatics and SWEBOK: a Way to Interconnection.* New Challenges of Economic and Business Development – 2016: Society, Innovations and Collaborative Economy. Abstracts of Reports. 2016. p. 125.

[12] Wheadon, M., Duval-Couetil, N. 2014. *Student Perspectives on Developing More Relevant Ph. D. Programs in STEM Disciplines through Professional Skills Training*. 2014 American Society for Engineering Education Annuel Conference, 24, 1. Paper ID # 10495.

TESMA: Towards the Development of a Tool for Specification, Management and Assessment of Teaching Programs

Nicolas Guelfi
University of Luxembourg
6 rue Coudenhove Kalergi
Luxembourg
+352.46 66 44 5251
nicolas.guelfi@uni.lu

Benjamin Jahić
University of Luxembourg
6 rue Coudenhove Kalergi
Luxembourg
+352. 691 30 74 38
benjamin.jahic.001@student.uni.lu

Benoît Ries
University of Luxembourg
6 rue Coudenhove Kalergi
Luxembourg
+352.46 66 44 5267
benoit.ries@uni.lu

ABSTRACT
Defining and managing teaching programs at university or other institutions is a complex task for which there is not much support in terms of methods and tools. This task becomes even more critical when the time comes to obtain certifications w.r.t. official standards. In this paper, we present an on-going project called TESMA whose objective is to provide an open-source tool dedicated to the specification and management (including certification) of teaching programs. This tool has been engineered using a development method called Messir for its requirements elicitations and introduces a domain-specific language dedicated to the teaching domain. This paper presents the current status of this project and the future activities planned.

Categories and Subject Descriptors
K.3.2 [**Computers and Education**]: Computer and Information Science Education – *accreditation, curriculum, self-assessment*.

General Terms
Documentation, Design, Languages.

Keywords
Teaching Programs Development, Tool-support, Software Engineering, Domain-Specific Languages, Automatic Generation.

INTRODUCTION
The University of Luxembourg is a young university (created in 2003). In this "start-up" context, we have been setting up new programs at bachelor, master and doctorate levels providing different education certificates. All those programs are offered to our students by three faculties. The need for a tool to support the task to define and manage (including certification) the education program came rapidly. The market analysis for this category of tools showed that no tool was available. A project has been started to engineer a method and a tool to support those needs. This project has been conducted following a software engineering process that had the following main steps:

- Requirements analysis: to provide the initial requirements for the TESMA tool, a requirement specification document has been produced using the Messir method [1].
- Design: to state the main choices concerning the TESMA architecture and interfaces.
- Implementation: to reach an operational system usable for validation w.r.t. the requirements.
- Those steps have been performed iteratively to produce the TESMA tool in an incremental way.

The content of this paper provides details on the requirements analysis, design and implementation of the TESMA tools. It also ends with a short related work section which summarizes what we have found when we did our market analysis.

1. TESMA REQUIREMENTS
Following part of the Messir method [1], we have elicited the actors that are concerned by teaching programs. They are:

- *The institution director* who represents the institution and validates new programs, courses and course modifications (e.g. the dean of a faculty, the head of a teaching unit, ...).
- *The program director* who specifies his programs and validates course modifications made by instructors.
- *The instructor,* who specifies, manages and maintains the courses he gives.
- *The student,* who tunes his curriculum (elective courses, ...) and receives information about his curriculum.

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

- *The secretary*, who is a delegate of any of the institutional actors (*institution director, program director or instructor)* and also ensures interoperability with other institution information systems.
- *The quality officer* who evaluates and validates the programs with respect to the universities internal laws. He's also responsible of program certification processes.

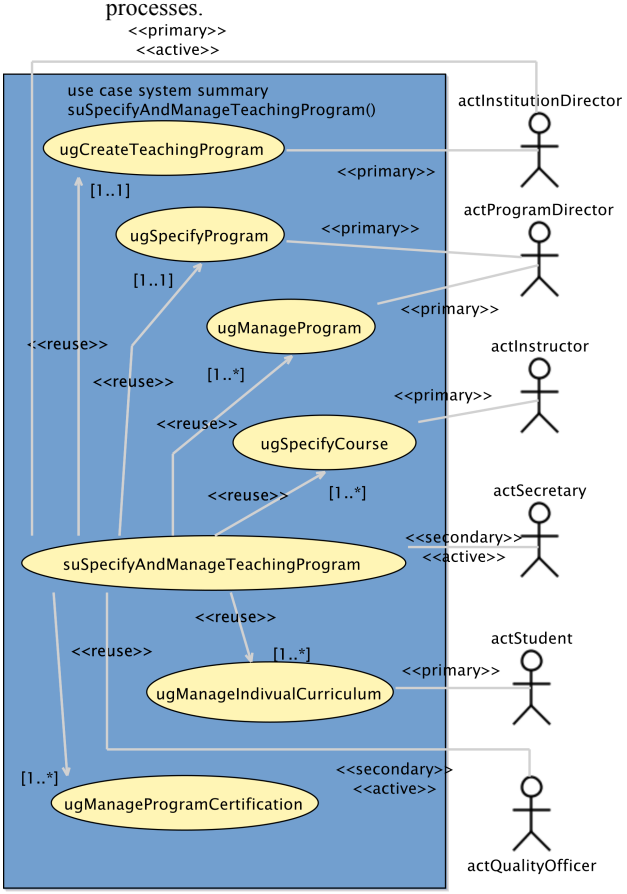


Figure 1. TESMA summary use-case.

You can find above a use-case model made in the context of the Messir method that displays the actors contributing to the high-level summary use-case dedicated to managing a teaching program.

The concepts managed by the TESMA actors are analysed and specified in the Messir concept model which is a UML class diagram. Among all the concepts that are necessary to specify the operations executed by the actors we have:

- concepts related to the actors and for which TESMA has to handle an internal representation: students, instructors, ...
- concepts related to the programs: course details, teaching periods, course evaluation, ...
- concepts related to program certification: standard description, standard coverage by an existing program, ...

In Messir those concepts are specified using an UML class diagram. Figure 2 provides a partial diagram view that models the concepts related to a program (institution, program, courses).

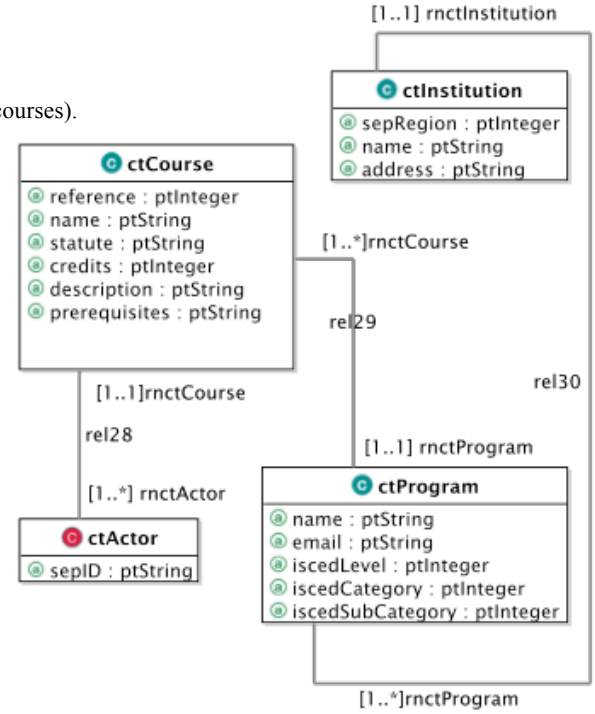


Figure 2. TESMA concepts.

The requirements analysis phase has allowed to determine a first version of the functionalities and data that should be handled in the first increment. The next section presents the design and implementation of this first version.

2. TESMA DESIGN AND IMPLEMENTATION

After having analysed the TESMA actors, concepts and functionalities. We have started to design a first version of the tool. A major design choice made is to allow the specification of programs using a domain-specific language (DSL) defined using the Xtext [2] framework. Thus, we designed TESMA as a plugin to the Eclipse workbench [3]. It is composed of three main architectural components illustrated in Figure 3 at the top of the diagram. Our components are based on stable Eclipse plugins themselves based on the Eclipse Modelling Framework (EMF) [4].

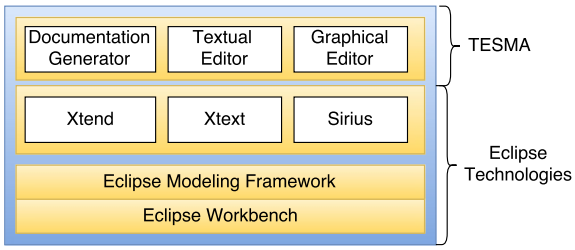


Figure 3. Architectural components overview.

2.1 Textual Editor

The main feature of the *Textual Editor* is to allow the specification of the teaching programs (this specification is called TESMA model in the remaining part of this paper) with the TESMA DSL. It also offers other supporting features, as for instance: syntactical validation rules, syntax highlighting, templates proposal, etc.

Xtext is an open-source framework that eases the development of domain-specific languages and offers features to provide a textual editor to the TESMA DSL. Xtext is based on EMF, which is the underlying-core library handling the TESMA model.

The TESMA DSL is designed to be intuitive, customizable and loosely coupled. In order to have an intuitive DSL, we have chosen to design its grammar using mainly keywords in natural language. Institutions may use different terminologies for the concepts used in our approach, this is why we designed the grammar of our DSL to be customizable. The institutions have the possibility to choose their own naming conventions. Lastly, the rules of the grammar are loosely coupled, i.e. optional cross-references are mostly used instead of containment relations.

2.2 Graphical Editor

The Graphical Editor provides a representation of the TESMA Model in a tabular view and offers the possibility to modify the TESMA model. The graphical editor provides typical table handling features like data sort, import/export from/to Excel sheets, hide/show columns, multiple rows selections.

The technology used to develop our graphical editor is Sirius [5], an open-source software Eclipse project that eases the creation of custom graphical modelling workbenches. Both Xtext and Sirius are based on EMF, which allows the TESMA tool-support to interact between Xtext and Sirius using EMF as underlying-core library for the TESMA model as represented in Figure 4.

Thanks to our tabular format, the graphical editor is intuitive and usable by non-computer experts. All the program's attributes are easy to access and modify. The modifications can be performed directly inside the graphical editor view.

2.3 Documentation Generator

The main feature of the *Documentation Generator* is to generate documents of different types, like Excel sheets, CSV files and PDF files. The *Documentation Generator* may be configured to produce a customized PDF file, e.g. by not generating some of the sections inside the pdf files.

The technologies used to develop the documentation generator are Xtend [2], Latex and the apache.poi library, for handling Excel sheets. Xtend is a programming language based on Java. It provides a compact syntax and eases the generation of natural language text. Latex is a document preparation system, which uses libraries, keywords and plaintext for writing scientific documents in pdf format. Finally, the apache.poi library provides the necessary tools for generating Excel sheets, which are used as teaching material.

The Documentation Generator has been designed to ease information retrieval in the generated Latex files. Additionally, it is designed to automatically update the final report, when the user manually adds data into the reserved appropriate folders. Finally, the different Latex files are imported inside one Latex file, which is compiled into a pdf file containing the program description.

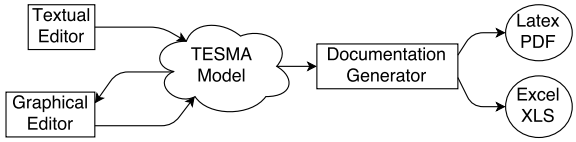


Figure 4. TESMA process overview.

3. ILLUSTRATION

We illustrate the TESMA approach with a course of a Master program named "Software Engineering Environment" (SEE) at the University of Luxembourg. Figure 5 is a screenshot of the TESMA tool-support in the Eclipse environment.

The TESMA model describing the SEE course have been specified using the approach described in this paper. The teaching program description of this example includes a number of textual files using the TESMA DSL syntax. The course run information is illustrated in Figure 5 by specifying the course teaching team organisation and dividing the teaching term into small periods and defining the tasks, tests for each period. The tasks and tests are defined in separate folders and referenced to the teaching period. At that point, TESMA is able to generate *a part of the Teaching Material*, like evaluation criteria, task lists and course information.

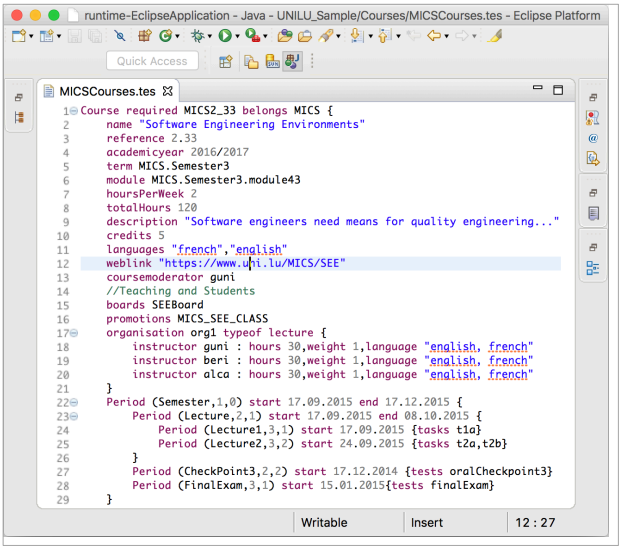


Figure 5. SEE specification in TESMA tool.

All other concepts can be specified using the TESMA DSL including the coverage of an education standard by an education program.

In this case study for the SEE course, we created for each category of TESMA model element a file containing all information related. In this case, one institution, one program and one course have specified, which represent about 10 textual files. We defined 10 instructors and 7 students for this example case, which are grouped in a single file. The total description in our case needs about 500 lines of specification text (>1000 in

case of certification). The specification text size mostly depends on the preciseness of the specifier. If the specification is done in details, the number of lines increases quickly. In general, it could vary from 100 to 1500 lines.

4. RELATED WORK

A number of related works have been performed in the past in the field of education programs and course specifications, especially for K-12 classes in high schools. However, we could not find methods, which are supported by tools, which help to design detailed teaching programs. On the one hand, some university guidelines are available online, e.g. [6,7], without offering supporting software applications. On the other hand, a few software applications are available, that do not offer comprehensive and customizable guidelines. We present in the following, three of these tools.

PDF Syllabus builder [8] is an open-source tool, which only offers a template PDF form for writing course syllabi. TESMA covers many additional features, for instance it generates automatically reports in PDF format. The design of the reports is standardized and generated by the TESMA tool.

Jump-Rope [9] is a proprietary tool, based on a web application. It supports a curriculum design tool, standards based gradebook, accurate attendance, administrator tools. TESMA has similar features and is more flexible in terms of customization of its textual language. Moreover, the use of MDE techniques allows the automatic reconfiguration of the user interface and the generated documents.

Build-Your-Own-Curriculum [10] is a proprietary tool, based on web-application developed for K-12 classes in the United States. It supports the feature of defining standardized courses, classroom managements, evaluations and assignments. TESMA is similar to this kind of tools, but has two major advantages: firstly, it is based on MDE technique, which allows customizing the documentation generation process in a flexible way. Secondly TESMA is an open-source project, which allows adapting freely its code to the institution's taste.

5. CONCLUSION AND FUTURE WORK

In this paper, we have shown how our project of engineering a tool for educational program and course specification based on a textual domain-specific language with a graphical editor has been developed. It generates documents out of the specification, which can be used by the institution's staff. Our tool and method has been successfully used on a small, yet real, example. As a future work we plan to iterate the process to stabilize the requirements and the tool design and implementation. We also plan to study the automated generation of a web application from the language grammar in order to provide a user-friendly front-end that is mapped to our textual language grammar.

6. REFERENCES

[1] Guelfi, N. 2016. *The Messir Scientific Approach to Requirements Engineering*. Laboratory for Advanced Software Systems Technical Report, TR-LASSY-16-01. University of Luxembourg.

[2] Bettini, L. 2013. *Implementing Domain-Specific Languages with Xtext and Xtend.* Packt Publishing Ltd.,Birmingham, UK.

[3] Beck, K., Gamma, E. 2003. *Contributing to eclipse – Principles, Patterns, and Plugins*. Addison-Wesley Professional.

[4] Steinberg, D., Budinsky, F., Paternostro, M., Merks, E. 2008. *EMF: Eclipse Modeling Framework, 2nd Edition*. Addison-Wesley Professional.

[5] Viyović, V., Maksimović M. and Perisić B. 2014. Sirius: A rapid development of DSM graphical editor. In *Proceedings of the IEEE 18th International Conference on Intelligent Engineering Systems* (Tihany, Hungary, July 3-5, 2014). INES 2014. IEEE, 233-238. DOI= http://dx.doi.org/10.1109/INES.2014.6909375.

[6] University of Washington. Course and Syllabus Design. Online, retrieved on July 2016. http://www.washington.edu/teaching/teaching-resources/preparing-to-teach/designing-your-course-and-syllabus/.

[7] Cornell University Center for Teaching Excellence. *Writing a Syllabus*. Online, retrieved on July 2016. http://www.cte.cornell.edu/teaching-ideas/designing-your-course/writing-a-syllabus.html.

[8] Joeckel, G., Longhurst, M. *PDF syllabus builder: open source tool for online instructors, course developers and instructional designers*. Online, retrieved on July 2016. http://olc.onlinelearningconsortium.org/effective_practices/pdf-syllabus-builder-open-source-tool-online-instructors-course-developers-and-i.

[9] Olsen, J., Meyer, J. *Curriculum Design Tool. Jump Rope*. Online, retrieved on July 2016. https://www.jumpro.pe/features/curriculum-design-tool/.

[10] School Software Group. *Build Your Own Curriculum (BYOC): Evaluating a K–12 online curriculum management system*. ProQuest. Online, retrieved July 2016. http://www.schoolsoftwaregroup.com/.

Lifetime Evaluation of Multi-Agent Predator-Prey System with Evident Biological Features

Yurij Lezhenin
Peter the Great St. Petersburg Polytechnic University
194021 St. Petersburg
Polytechnicheskaya, 21
Russian Federation
yurij.lezhenin@gmail.com

Artyom Zhuikov
Peter the Great St. Petersburg Polytechnic University
194021 St. Petersburg
Polytechnicheskaya, 21
Russian Federation
artem.zhuikov@mail.ru

Vadim Dyachkov
Peter the Great St. Petersburg Polytechnic University
194021 St. Petersburg
Polytechnicheskaya, 21
Russian Federation
vadik.dyachkov@gmail.com

ABSTRACT

Artificial life systems attract the attention of researchers due to their ability to model and study natural processes by means of software and hardware simulating environments. One of these is the predator-prey concept which is widely used for modeling the relations in ecosystems where the members have contradictory goals. The article suggests a software model of multi-agent predator-prey system. We engaged the obvious biological principles to construct the model parameters laws of variation. The goal of the article is to evaluate the model lifetime dependencies. The experiments aimed at obtaining the best ratios of predators-prey populations and the grass growing intervals were performed. A balance between random and non-random model components has been estimated. The macro-parameters' combinations returning the mean lifetime value overcoming 10-20 times the values obtained when using macro-parameters set at random have been defined. It has been also found that the grass growing amount and interval affect the vitality of the model, consequently, the hypothesis of optimal feeding ratio has been inferred.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence

General Terms

Algorithms, Performance, Experimentation

Keywords

Artificial Life, Predator-prey, Software modeling

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

1. INTRODUCTION

Artificial life systems provide a powerful tool for studying biological evolution phenomena by means of numerical experiments. Artificial life models have been in focus of the researchers since the middle of the 20-th century. Since then the main concepts in this area have been introduced and a number of approaches has been applied to software modeling of ALife systems, such as game theory [2], [6] optimization including genetic algorithms [4], neural networks [1], search algorithms [5] etc.

Despite the extensive ALife research and numerous instruments for simulation we developed from scratch a custom lightweight and fast ALife implementation which makes possible a large number of statistical experiments using ordinary computational device. Also for future work and collaboration the model code is open-source and extendable. The article suggests a software model of multi-agent predator-prey system. We used the obvious biological assumptions to choose the model parameters analytical description. The elementary model of a single-agent (an animal) is based on a number of features typical for natural predator-prey systems: age; relative velocity and vision; reproduction mechanism; nutritive value of food; energy and stamina. The agents are driven by their "brain", they have four types of behavior and make the decisions based on the state of the environment. The goal of the research presented in this article is to simulate the predator-prey world, to study the lifetime dependencies of the model discussed and to keep it extensible and agile.

2. MODEL

The multi-agent predator-prey artificial life system is considered to have three agent types: predators, preys and grass. The model derivation process includes the construction of a:

1. Single agent model, i.e. agent life processes model.
2. Multi-agent model.

2.1 Single agent model

It is assumed in the research that predators and preys are described by a number of temporary evolving parameters (Eq. 1): age (Eq. 3); velocity (Eq. 4); life energy (Eq. 5); stamina (Eq. 7); vision (Eq. 9). At the same time, grass is characterized by only one parameter (Eq. 2). A number of steps t is taken by the authors as an independent argument, while the model parameters depend upon it directly

or implicitly.

The agent model is defined by:

$$\{Animal\}_{i=1}^{N} = \{A_i, V_i, E_i, S_i, R_i\}) \quad (1)$$

$$\{Grass\}_{i=1}^{M} = \{A_i\} \quad (2)$$

$$A = \alpha \cdot t \quad (3)$$

$$V \leq V_{max} = \beta \cdot V_0 \quad (4)$$

$$E \leq E_{max} = \lambda \cdot E_0, \quad (5)$$

$$\Delta E = E(t) - E(t-1) = \gamma \cdot E_{food} - \epsilon \cdot E_{life} - \zeta \cdot E_{move} \quad (6)$$

$$S \leq S_{max} = \theta \cdot S_0 \quad (7)$$

$$\Delta S = S(t) - S(t-1) = \eta \cdot S_{move}, \quad (8)$$

$$R = \rho \cdot R_0, \quad (9)$$

where ζ, η are covered distance-dependent linear coefficients, λ, β, γ, ϵ, ρ are age-dependent polynomial coefficients, θ is energy-dependent exponential coefficient and α, V_0, E_0, E_{food}, E_{life}, E_{move}, S_0, S_{move}, R_0 are constants (see Appendix 1).

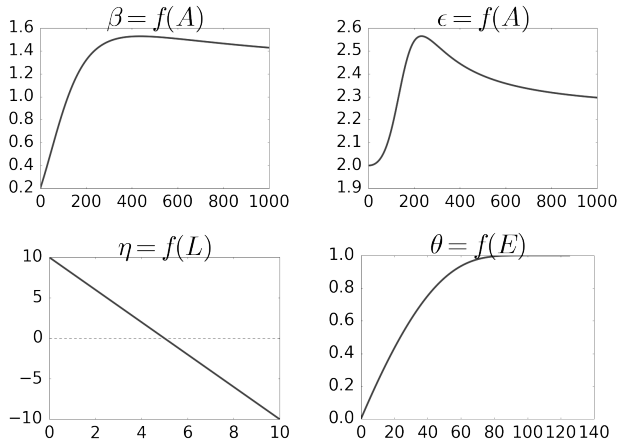


Figure 1: Model coefficients

2.2 Multi-agent model

Multi-agent modelling is performed in a rectangular closed area of $l \times w$ size with cells, where l is the area length; w is its width. Each agent occupies one cell. It moves on a number of cells and it depends on the velocity. The world (W) of multi-agent model consists of:

$$W = \{\{Animal\}_{i=1}^{N_D}, \{Animal\}_{j=1}^{N_Y}, \{Grass\}_{k=1}^{N_S}\} \quad (10)$$

where $N_D = N_{preDators}$, $N_Y = N_{preYs}$, $N_S = N_{graSs}$ are partial amounts of each type of agents measured in cells.

A global energy over model is determined as the sum of life energy of predators and preys:

$$E_W = \frac{(E_D + E_Y) \cdot E_D \cdot E_Y}{1 + E_D \cdot E_Y} \quad (11)$$

where $E_D = \sum_{i=1}^{N_D} E_{preDator}$, $E_Y = \sum_{i=1}^{N_Y} E_{preYs}$ are agents life energy. The modeling process stops when $E_D = 0$ and/or $E_Y = 0$.

The lifetime Z of the model world W is measured as the number of steps during which the model stays alive $(E_W > 0)$.

2.3 Model behavior control

The predators and preys have "brain" and make decisions, while grass behavior is direct: a fixed amount of grass appears at random positions within a fixed number of steps. The decision making system comprises walking, eating, reproduction and pursuit/runaway.

2.3.1 Brain

At each step animal makes decision based on a target list and its actual energy level is:

$$k = \frac{E(t)}{E_{max}}. \quad (12)$$

Every target on the list is a structure [A target type, distance to target]. This way each agent gets a snapshot of the environment state around him.

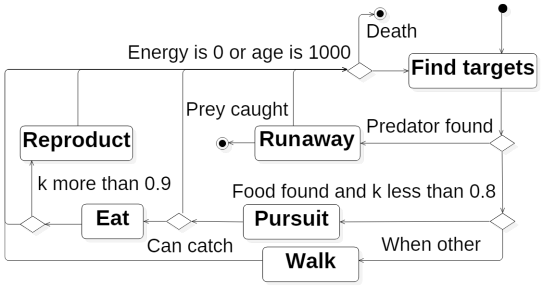


Figure 2: Prey brain

2.3.2 Behavior

Four types of agent behavior are considered:

1. **Walking** as a free agent moving driven by C++ pseudo-random generator. During walking the animal spends the moving energy $\zeta \cdot E_{move}$ (see Eq.6) and restores its stamina $\eta \cdot S_{move}$ (see Eq.8).
2. **Eating** as a food consumption process that gives the energy amount $\gamma \cdot E_{food}$ (see Eq. 6). Eating action is performed within a current step, i.e. instantly.
3. **Reproduction** as a new agent insertion into the model. After that the energy level k of a parent is reduced up to 0.6
4. **Pursuit or runaway** as moving with maximal available velocity $\beta \cdot V_0$ (see Eq.4) towards prey or from predator along the line connecting their initial positions. During pursuit or runaway the animal spends the moving energy $\zeta \cdot E_{move}$ (see Eq.6) and looses its stamina $\eta \cdot S_{move}$ (see Eq.8)

3. EXPERIMENTAL ENVIRONMENT AND METHODOLOGY

3.1 Environment

The experimental environment is formed by 2 parts: C++ model implementation and the research part deployed on Jupyter, an interactive data science and scientific computing web-application (Fig. 3).

The source files of the model can be accessed on GitHub[1] and can be compiled. C++ implementation was motivated by a high performance and efficiency of the language. At

[1]https://github.com/Zhuikov/predator-prey

the same time the object-oriented paradigm seems to be relevant to predator-prey models. The development methodology driven by unit and functional tests was used.

Our application is multi-platform (tested under Windows 10, Debian 8.5) and is compiled with Qt Framework.

The model launch parameters, data analysis and visualization of the results for the model testing and experiments are set by Python scripts in Jupyter notebooks. Jupyter, NumPy, SciPy and matplotlib packages are installed by the Conda package management system and the environment file is also available. A Docker container `zhuikov/predator-prey:20160702` on DockerHub is provided. All the components of experimental infrastructure and original artifacts are available and the results can be easily repeated, replicated or reproduced [3]. C++ application and Jupyter notebooks are launched in continuous integration system Travis CI, the travis configuration file can be accessed. The tools summary is placed in Appendix B.

3.2 Model implementation

The model implementation consists of the model core, graphical user interface (GUI) and command line interface (CLI) (Fig. 3). The model core implements agents, decision making and behaviour described in Sec. 2. GUI was implemented with Qt Framework 5.7. It allows model behavior observation in real-time and parameters tuning. Data acquisition in CSV format is performed through CLI.

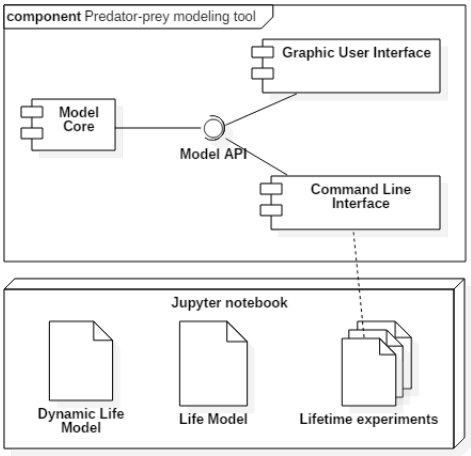


Figure 3: Component diagram

3.3 Experiment

The goal of the experiment was to search for macro-parameters combinations that give the longest model lifetime:

$$f : \{\mathbb{N}_{\mathbb{D}} \times \mathbb{N}_{\mathbb{Y}} \times \mathbb{G}_{\mathbb{N}} \times \mathbb{G}_{\mathbb{I}}\} \to \mathbb{Z} \tag{13}$$

where $\mathbb{N}_{\mathbb{D}}, \mathbb{N}_{\mathbb{Y}}, \mathbb{G}_{\mathbb{N}}, \mathbb{G}_{\mathbb{I}}$ are four vectors of model macro parameters: a number of predators and preys; grass amount and growing interval respectively.

3.3.1 Parameter settings

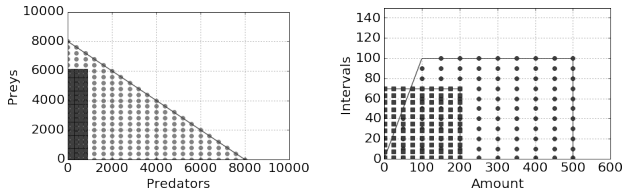
A single-agent model was tuned under the following assumptions:

- The area size is 100×100 cells.
- The grass appears at $\mathbb{G}_{\mathbb{N}}$ cells with $\mathbb{G}_{\mathbb{I}}$ steps interval.
- The agent lifetime limit is set to 1000 steps.
- The age of the first generation of the animals is set to 400 steps.
- The initial energy level is set to a starting level and holds on for 50 steps to keep young animals alive.
- The stamina decreases rapidly when energy factor $k < 0.6$.

The model constant parameters has been empirically set. They are summarized in Appendix 1.

3.3.2 Experiment plan

The experiments were performed within regular plans, where the elements of $\mathbb{N}_{\mathbb{D}}$ and $\mathbb{N}_{\mathbb{Y}}$ cover the range of $1-8000$ (triangle Fig 4(a)). Then the simulation step was refined and the experiments were repeated on a part of plan points, where the maxima were discovered (rectangle Fig 4(a)). The elements of $\mathbb{G}_{\mathbb{N}}$ and $\mathbb{G}_{\mathbb{I}}$ varied in accordance with the trapeze (Fig. 4(b)), after that the experiment was repeated at the plan points inside the rectangle at Fig. 4(b).



(a) (b)

Figure 4: Experiment plan: (b) $\mathbb{N}_{\mathbb{D}}$ and $\mathbb{N}_{\mathbb{Y}}$; (a) $\mathbb{G}_{\mathbb{N}}$ and $\mathbb{G}_{\mathbb{I}}$

4. RESULTS AND DISCUSSION

The model was run once at each plan point and the lifetime values $Z_i \in \mathbb{Z}$ it returned were stored. One can notice, that those plan points $\{N_D^i, N_Y^i, G_N^i, G_I^i\}$ where the lifetime local maxima were observed are located in the vicinity of each other. For all local maxima points the grass growing amount versus growing interval ratio was near $5:2$ and the predators/preys ratio was in the range from $1:200$ to $1:4000$. The model was launched again only for those plan points where these ratios were held.

The model launch around maxima with finer simulation step size resulted in ridge-like peaks of the function $\mathbb{Z} = f(\mathbb{N}_{\mathbb{D}}, \mathbb{N}_{\mathbb{Y}})$ for $G_N = 50$ and $G_I = 20$. The ridges stretch along the line of best ratios (Fig. 6(a)), and $\mathbb{Z} = f(\mathbb{G}_{\mathbb{N}}, \mathbb{G}_{\mathbb{I}})$ with fixed $N_P = 1$ and $N_Y = 1200$ (Fig. 6(b)).

Since the agents' positions were randomly defined, the lifetime breakdowns for best ratios and for arbitrary ones were explored. The results got from model launched with different random seeds allow to deduce that best ratios return the mean lifetime value that overcomes 10–20 times the values obtained using macro-parameters set at random (Fig. 7(a), 7(b)).

5. FUTURE WORK

The group pursuit strategies and search algorithms as well as amphigenetic agents reproduction may be applied and implemented. Genetic algorithms seem to be fruitful for further parameters tuning and evolution modeling.

6. CONCLUSIONS

The multi-agent Alife model to explore predator-prey system has been suggested. The authors used biologically rea-

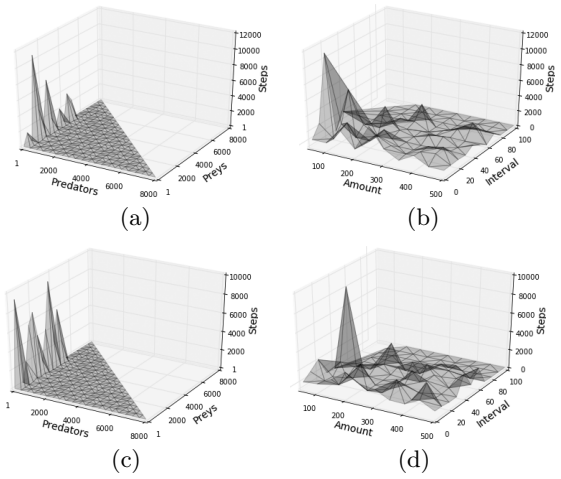


Figure 5: Lifetime dependencies: (a) $\mathbb{Z} = f(\mathbb{N}_\mathbb{D}, \mathbb{N}_\mathbb{Y})$ for $G_N = 50$ and $G_I = 20$; (c) $\mathbb{Z} = f(\mathbb{N}_\mathbb{D}, \mathbb{N}_\mathbb{Y})$ for $G_N = 100$ and $G_I = 40$; (b) $\mathbb{Z} = f(\mathbb{G}_\mathbb{N}, \mathbb{G}_\mathbb{I})$ for $N_P = 1$ and $N_Y = 1200$; (d) $\mathbb{Z} = f(\mathbb{G}_\mathbb{N}, \mathbb{G}_\mathbb{I})$ for $N_P = 1$ and $N_Y = 3600$.

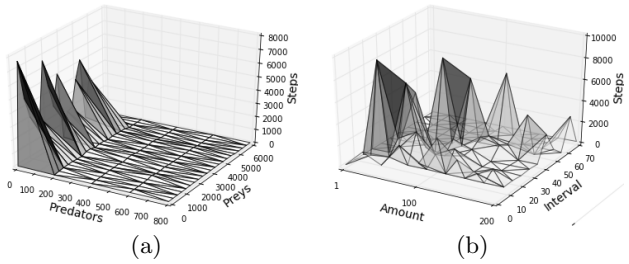


Figure 6: Best ratio ridges: (a) $\mathbb{Z} = f(\mathbb{N}_\mathbb{D}, \mathbb{N}_\mathbb{Y})$ for $G_N = 50$ and $G_I = 20$; (b) $\mathbb{Z} = f(\mathbb{N}_\mathbb{D}, \mathbb{N}_\mathbb{Y})$ for $G_N = 100$ and $G_I = 40$.

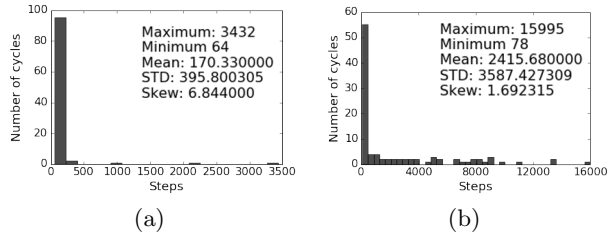


Figure 7: Lifetime breakdowns (a) for $\frac{N_D}{N_Y} = 400 : 2700$ and $\frac{G_N}{G_I} = 75 : 14$ (b) for $\frac{N_D}{N_Y} = 1 : 1800$ and $\frac{G_N}{G_I} = 25 : 14$

sonable assumptions for the agent and world models construction. The model is defined as a set of functions of multiple variables which allows to find lifetime extremes. The programming code is extensible and can be easily accessed for further experiments and improvement. The relations between the numbers of agents and grass parameters have been searched for and some stable ratios when model lives 10–20 times longer than at arbitrary chosen parameter settings were found.

7. REFERENCES

[1] G. Beuster. *Using Neural Networks and Genetic Algorithms as Building Blocks for Artificial Life Simulations*, pages 363–366. Springer Vienna, Vienna, 2001.

[2] K. Dutta. Hunting in groups. *Resonance*, 19(10):936–957, 2014.

[3] D. G. Feitelson. From repeatability to reproducibility and corroboration. *SIGOPS Oper. Syst. Rev.*, 49(1):3–11, Jan. 2015.

[4] S. Ito, Pilat and Arita. Population and evolutionary dynamics based on predator-prey relationships in a 3d physical simulation. *Artificial Life*, 22(10):226–240, 2016.

[5] R. E. Korf. Heuristic evaluation functions in artificial intelligence search algorithms. *Minds and Machines*, 5(4):489–498, 1995.

[6] H.-Y. Shi, W.-L. Wang, N.-M. Kwok, and S.-Y. Chen. Game theory for wireless sensor networks: A survey. *Sensors*, 12(7):9055, 2012.

APPENDIX

A. MODEL CONSTANTS

Table 1: Parameters

Par.	Description	Value
α	Age scale coef.	0.007
β	Velocity coef.	$\frac{A^2+2\cdot A}{A^2+2.3} + 0.2$
V_0	Baseline velocity	2
λ	Energy coef.	$\frac{A^2+2\cdot A}{A^2+3}$
E_0	Baseline energy	100
γ	Food value coef.	$\frac{5\cdot A^2+5\cdot A}{7\cdot A^2+3}$
E_{food}	Food energy	165
ϵ	Life coef.	$\frac{A^3}{(4.5\cdot A^3)-(14\cdot A^2)+18\cdot A-0.1} + 2$
E_{life}	Life energy	1
ζ	Movement coef.	$0.2 \cdot L$
E_{move}	Movement energy	1
S_{move}	Movement stamina	10
η	Movement coef.	$-2 \cdot \frac{V}{V_{max}} + 1$
S_0	Baseline stamina	50
θ	Stamina coef.	$1 - e^{\frac{-3\cdot E(t)}{E_{max}-E(t)}}$
R_0	Baseline vision	8
ρ	Vision coef.	$\frac{A^2+2\cdot A}{A^2+2.3} + 0.2$

B. TOOLS

Table 2: Tools

Operating systems	Debian 8.5, Windows 10
C++ Framework	Qt Framework 5.7.0
Data science and scientific computing tool	Jupyter 4.1.0, matplotlib 1.5.1, NumPy 1.11.0, SciPy 0.17.1
Software containerization platform	Docker 1.8.3
Continuous integration system	Travis CI

Formal Verification of Technical Systems using smartIflow and CTL

Philipp Hönig
Hochschule Ulm
University of Applied Sciences
Institute of Computer Science
hoenig@hs-ulm.de

Rüdiger Lunde
Hochschule Ulm
University of Applied Sciences
Institute of Computer Science
r.lunde@hs-ulm.de

Florian Holzapfel
TU München
Institute of Flight System Dynamics
florian.holzapfel@tum.de

ABSTRACT

Verification of safety requirements is one important task during the development of safety critical systems. The increasing complexity of systems makes manual analysis almost impossible. This paper introduces a methodology for formal verification of technical systems with smartIflow. Existing approaches mainly use existing model checking tools for this task. Due to the bidirectional connection modeling, this is not feasible for smartIflow models. Our proposed two-step method first predicts the system behavior. In the second step, safety requirements specified in CTL are verified, and counterexamples are generated if these are not satisfied. We describe the usage of CTL formulas verified against safety critical complex systems modeled with smartIflow. The practical applicability is shown using a simple example system.

Categories and Subject Descriptors

B.8.1 [**Hardware**]: Reliability, Testing, and Fault-Tolerance; I.6.2 [**Computing Methodologies**]: Simulation Languages

General Terms

Algorithms, Reliability, Languages

Keywords

Model-Based Safety Analysis, smartIflow, FSM, DES, Model Checking, CTL, LTL

1. INTRODUCTION

During development of safety critical systems serveral analysis tasks like FMEA (Failure Mode and Effects Analysis), FTA (Fault Tree Analysis) or CCA (Common Cause Analyse) are performed [5]. Besides that, safety engineers often verify the correctness of systems using safety requirements specifications. Performing this task manually can be time consuming and error prone since every system reaction to failures or external inputs have to be predicted. SmartIflow (State Machines for Automation of Reliability-related Tasks using Information FLOWs) [7] has been designed to automate the safety analysis process. The modeling formalism behind smartIflow is on a quite high level of abstraction. Due to the bidirectional connection modeling and the flexible property mechanism, the predictive power of smartIflow is quite high. Our previous experiments with smartIflow have shown that the level of abstraction offers a good compromise between computational effort and predictive power. However, in these experiments only specific scenarios (e.g. reaction to a broken pipe) were simulated. Existing approaches like AltaRica3 [3] or deviation models[11] use a similar level of abstraction, however, with less expressiveness (e.g. there are no built-in elements for flow direction determination).

This paper describes an approach for automated verification of technical system with smartIflow. The objective is to automatically verify a safety specification against a system model. In that context the following questions arise:

- How can such requirements be described in a formal language?
- How can the simulation be controlled to capture all relevant sequences of events?
- How might a verification result look like?

We will answer these questions by introducing our new verification method which is based on temporal formulas. It is being studied whether and how model checking techniques can be used.

This paper is organized as follows. The following section briefly summarizes the modeling concepts behind smartIflow. In Section 3 some fundamental model checking concepts are described. Existing approaches supporting formal verification of safety requirements are described briefly in Section 4. In Section 5 the automated verification of safety requirements with smartIflow will be explained. The results of our experiments with a small example system are shown in Section 6. Finally a short conclusion will be given.

2. THE SMARTIFLOW FORMALISM

The modeling formalism behind smartIflow is object- and component-oriented. Figure 1 visualizes the component model of a 3-position valve. Basically, each component in a system is considered as finite state machine. Thus a component consists of a set of state variables that are used to capture the operational and failure modes of a component. State changes are either performed after events or signal changes on ports initiated by other components. Events are used

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

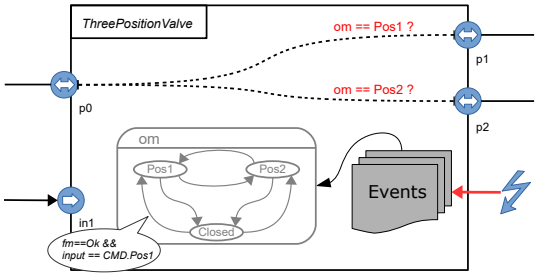


Figure 1: Sketch of a smartIflow component

to stimulate a system externally, for example to change the operational mode of a component. Ports represent the connection points of a component. There are two groups of port types, namely logical ports (e.g. input or output), and physical ports (e.g. for undirected electrical connections). Components are linked through these ports. The state-dependent behavior is described in terms of modification of the connection structure and also by property propagation through the network. Actually, properties are key-value pairs that provide a quite flexible mechanism to abstract from physical flow information. Connections are modeled as undirected channels in case of physical conductors, and there are also unidirectional channels for logical signals. Built-in primitives enable flow direction determination. The information about flow direction is available at each port by means of a reserved property (e.g. *flow.dir=IN*). Models in smarIflow can be composed graphically in Simulink/Simscape as we showed in our previous work [8].

3. MODEL CHECKING

Model Checking [2] describes the process of verifying a system model M against a specification ϕ. The objective is to proof whether M fulfills the specification ϕ. In other words: $M \vDash \phi$? This verification is performed fully automatically. System models describe the possible system behavior in a formal way. Finite state machines or transition systems are often used for behavioral description. The verification algorithm systematically explores all states and tries to disprove the specification. If the model doesn't fulfill the specification, the algorithm will deliver a counterexample, i.e. a trace of the system behavior that falsifies the property. However, most model checking tools like NuSMV deliver only one counterexample even though there could exist more [1].

3.1 Linear Temporal Logic

Most model checking tools allow specifications to be expressed in temporal logics, like LTL or CTL. LTL (Linear-time temporal logic) [9] allows to create formulas about future paths. Consequently, besides the usual logical operators ($\neg, \vee, \wedge, \Rightarrow$ *and* $\Leftrightarrow$) and operators for atomic expressions ($==, !=, <=, ...$) there is also a set of temporal Operators:

- $X\phi$: ϕ has to hold in the ne**X**t state
- $G\phi$: ϕ has to hold at each state of the subsequent path (**G**lobally)
- $F\phi$: ϕ has to hold somewhere on the subsequent path
- $\psi U\phi$: ψ has to hold **U**ntil ϕ holds
- $\psi R\phi$: ϕ has hold up to the moment when ψ becomes true (**R**elease)

An example for the safety specification "Every occurrence of an event is eventually followed by an action" expressed in LTL might look like as follows: G(event $\Rightarrow$ F(action)).

3.2 Computation Tree Logic

In contrast to LTL, CTL (Computation Tree Logic) allows to create statements about states of trees. Therefore CTL provides besides logical operators a set of temporal connectives. Each temporal connective is a pair of symbols. The first symbol specifies the path quantifier which is either A (along **A**ll paths) or E (there **E**xists at least one path). The second symbol stands for a set of temporal operators namely X (at the ne**X**t state), F (at a **F**uture state) and G (at all future states, i.e. **G**lobally). The combination of these two symbols defines the most important operators (EX, EF, EG, AX AF and AG) to specify properties that take into account the non-deterministic behavior of a system. Consider the following property: "It's always possible that state P occurs and holds for the rest of time". This may be expressed in CTL with the formula AF(EG(P)). Figure 2 shows two example systems that fulfills $AF\phi$ respectively $AG\phi$

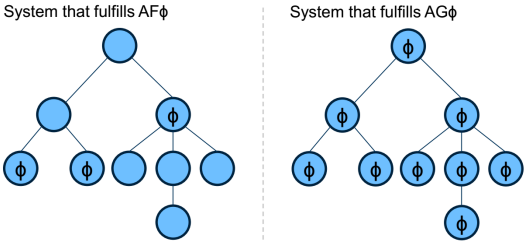


Figure 2: CTL: Examples

The expressiveness of LTL and CTL is quite different. There is an overlap, however, there are statements that can be expressed in CTL but not in LTL and vice versa. One should note that for instance NuSMV cannot generate counterexamples for all kind of CTL specifications [1].

4. RELATED WORK

Most approaches towards formal verification of safety requirements that can be found in literature utilize temporal logic and existing model checking tools. Joshi et al. proposes an approach based on Simulink and NuSMV model checker [10]. System models created in Simulink are extend with an fault model by using failure effect modeling. This so called extend system model is translated into the input language of the NuSMV model checker. In principle, Simulink is just used as alternative (graphical) representation for NuSMV models. Safety requirements specification created in CTL are used to verify the system behavior. If the system does not fulfill the specification, one counterexample with a trace of states that violated the specification is created.

The SLIM (System Level Integrated Modeling Language) language was developed by the COMPASS (Correctness, Modeling project and Performance of Aero space Systems) project for modeling hardware and software systems for safety-related tasks [4]. Their framework supports several analysis methods, among others, generation of FMEA tables, fault tree analysis, and correctness verification. Again, NuSMV is used as fundamental platform for the various analysis tasks. System models in SLIM are translated into the input lan-

guage of NuSMV and temporal logic is used for requirements specification.

Another approach based on model checking has been introduced by Güdemann et al. [6]. System models are constructed in SAML (Safety Analysis and Modeling Language) which can be translated in several analysis tools like NuSMV or PRISM. This approach allows both, qualitative and quantitative analysis.

5. FORMAL VERIFICATION OF SAFETY REQUIREMENTS

As described in the previous section, most approaches utilize existing model checking tools like NuSMV. This is obviously not feasible for smartIflow models, among other, due to the bidirectional connection modeling and flow direction determination. The input languages of existing model checkers don't provide such features, and because of that using existing model checker by translating smartIflow models in the input language of them is not possible. For this reason, we developed a model checking algorithm for smartIflow. Most model checking tools explore the system behavior depending on a requirements specification and try to disprove the specification. In our approach this is split up into two parts. First, the system behavior is simulated, and in the second step the specification is verified. This uncoupling has the great advantage that the verification is not dependent on the kind of simulation. Conversely, various specifications can be checked without re-simulating the system. However, this two-step approach may lead to problems when trying to keep search space small (e.g. only predict nominal behavior), because in that case we cannot profit from the specification. Both steps are now described in detail below.

5.1 Behavior Prediction

```
s = initial system state;
Q = ∅; // Open state nodes
Result = ∅; // Processed state nodes
add(Q,s) ;
while Q not empty do
    u = Q.remove() // Remove first item
    reconfigure network according to state(u);
    determine flows and propagate properties;
    v = compute possible next states;
    foreach possible next state e in v do
        if e not in Result then
            Q.add(e);
            Result.add(e);
        else
            create reference to existing state node;
        end
        if executeEvent(e) then
            next-states = execute admissible events;
            Q.addAll(next-states);
        end
    end
end
```

Algorithm 1: Behavior Prediction

Algorithm 1 describes the fundamental steps for behavior prediction. After a initialization phase, where all state variables are set to their default value, the subsequent system states can be predicted. A single simulation step basically consists of four substeps. During the **network reconfiguration**, connections are created and properties are published depending on the values of the state variables. After that, **flow directions are determined** and properties are propagated. Thereafter, **state variables are updated** depending on the current state variable values and port properties. In case of non-deterministic transitions, the result of this substep will be a set of subsequent system states. System states that already have been simulated previously get a reference to the existing state. Therefore, such states don't have to be simulated further, which reduces the computational effort, and also the total number of system states. Since we are interested in all possible system reactions on failures or input events, we have to stimulate the system using **events** at certain system states. We cannot execute events of a system in all possible combination at each state since this would lead to state-space explosion. This is also completely unnecessary, because there are a lot of constellations which can not occur in reality (e.g. several events at the same time). For this purpose we have developed a formal language that enables to describe the permissible combinations of events at a state node. For instance, events can be restricted to a special type or the number of events of a specific type (e.g. failure events) can be limited. Due to space limitations we cannot go into detail on the syntax and semantics of the language. Events are triggered only in certain states (e.g. stable state or alternating state). Therefore for each new expanded state node it is first checked whether events shall be triggered at all. After that, all possible events with respect to event specification are added to the state node and the possible subsequent states are determined.

A simulation result is a directed graph in which each node corresponds to one system state. Nodes reference other nodes, states do not reference states. Transitions which are caused by external events are labeled with a unique identifier of the event. Figure 3 shows a possible outcome of behavior prediction.

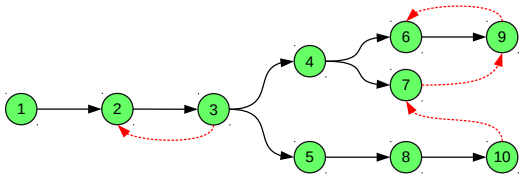


Figure 3: Simulation Result

5.2 Requirements Verification

The safety requirements need to be specified in a formal language. Typical safety requirements for a technical system look like as follows:

- "It is always possible to reach state X"
- "After pressing switch X must necessarily follow the action Y".

In case of the first requirement, LTL is not adequate since LTL can only express that X is actually reached and not that it can be reached. As already described in Section 3.1, LTL only allows to make specifications on a single path. CTL can state this property with formula $AG(EF(X))$. Therefore we decided to use CTL. Specifications can be created using all usual CTL operators ($AG, AF, AX, ...$). In atomic formulas, comparisons between all kind of variables including propagated properties and symbolic values can be expressed. Both, properties at ports and variables can be ac-

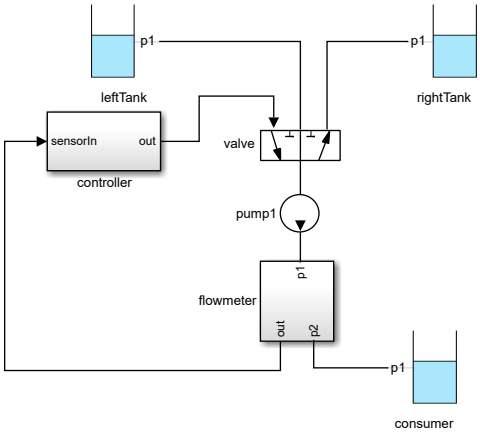


Figure 4: Example System

cessed via the absolute path, starting at root component (e.g. $MAIN.valve1.port1.flow.dir == IN$).

The evaluation of path quantifier in CTL formulas is in principle based on graph exploration. Depending on the operator, an expression ϕ is for instance verified at all paths at each node (in case of operator AG). ϕ can be any kind of expression, even an expression with a path quantifier. If ϕ is an expression with a path quantifier, the evaluation of the expression does not begin at root node of the simulation result, but rather at the current position of exploration. In case of a violation of a CTL formula, a counterexample is generated. A counterexample is characterized by a trace of events that have been executed on the path from the root node to the node where the specification is violated. Unlike NuSMV, which only generates one counterexample, our approach is able to create all counterexamples that violate a specification. However, there are still CTL expressions that will not return any counterexample. For example the expression $AG\phi \wedge AG\psi$ will just result to *TRUE* or *FALSE*. Generation of counterexamples is not possible in this case, since this would lead to ambiguities.

6. EXAMPLE

Figure 4 shows a very simple example system. The system consists of three tanks (left, right, consumer), a 3-position valve, flow meter, pump and a controller. The controller is responsible for keeping the consumer always supplied with liquid. In case of an empty tank (established through the flow meter), the controller is able to switch to the second tank. A safety requirements specification for this system may look like as follows: AG((((MAIN.leftTank.fm==Ok || MAIN.rightTank.fm==Ok) && MAIN.flowmeter.fm != Leakage) → AF(MAIN.consumer.p1.flow.dir == IN))). This means that at least one tank must be functional (no leakage, not empty), and there must be no leakage at flow meter, eventually the consumer is supported with fluid. Obviously this property is not satisfied. Consequently verification algorithm will deliver a quite comprehensive set of counterexamples. For instance the following event sequence will falsify the property: MAIN.v1.ActivateStuckAtPosition2 → MAIN.right.SetEmpty; The valve stuck at position two (supply from the right tank) and after that the right tank is empty. The controller tries to change the supply from the first tank, however the valve cannot changes the position. Therefore the consumer will no longer be supported with fluid.

7. CONCLUSION

In this work we described a method of formal verification of technical systems using smartIflow. Existing approaches to formal verification mostly use existing model checking tools. We cannot utilize these tools since the modeling formalism of smartIflow is quite different to input languages of existing model checkers. Despite the fact that we cannot use existing model checking tools, we use the powerful temporal logic CTL to specify the safety requirements. In contrast to model checking tools like NuSVM, our approach enables generation of multiple counterexamples. In the next step, we plan to enrich the events with information about the probability of failure to calculate the total probability of failure.

8. REFERENCES

[1] *NuSMV 2.5 User Manual*. Italy, 2010.

[2] C. Baier and J.-P. Katoen. *Principles of Model Checking (Representation and Mind Series)*. The MIT Press, 2008.

[3] M. Batteux, T. Prosvirnova, A. Rauzy, and L. Kloul. The altarica 3.0 project for model-based safety assessment. In *2013 11th IEEE International Conference on Industrial Informatics (INDIN)*, pages 741–746, July 2013.

[4] M. Bozzano, A. Cimatti, J.-P. Katoen, V. Y. Nguyen, T. Noll, and M. Roveri. *The COMPASS Approach: Correctness, Modelling and Performability of Aerospace Systems*, pages 173–186. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[5] Federal Aviation Administration (FAA). *FAA System Safety Handbook, Chapter 9: Analysis Techniques*, Dezember 2000.

[6] M. Gudemann and F. Ortmeier. A framework for qualitative and quantitative formal model-based safety analysis. In *High-Assurance Systems Engineering (HASE), 2010 IEEE 12th International Symposium on*, pages 132–141, Nov 2010.

[7] P. Hönig and R. Lunde. A new modeling approach for automated safety analysis based on information flows. In *25th International Workshop on Principles of Diagnosis (DX14)*, Graz, Austria, 2014.

[8] P. Hönig, R. Lunde, and F. Holzapfel. Modeling technical systems with smartiflow for safety related tasks. In *International Workshop on Applications in Information Technology (IWAIT-2015)*, Aizu-Wakamatsu, Japan, 2015.

[9] M. Huth and M. Ryan. *Logic in Computer Science: Modelling and Reasoning About Systems*. Cambridge University Press, New York, NY, USA, 2004.

[10] A. Joshi, M. Whalen, and M. P. Heimdahl. Modelbased safety analysis: Final report. Technical report, University of Minnesota, 2005.

[11] P. Struss and S. Dobi. Automated Functional Safety Analysis of Vehicles Based on Qualitative Behavior Models and Spatial Representations. In *The 24th International Workshop on Principles of Diagnosis (DX-2013).*, pages 85–91, 2013.

Adopting Tree Overlapping Algorithm for MathML Equation Structural Similarity Evaluation

Mikhail Ponomarev
Peter the Great St. Petersburg Polytechnic University
29 Polytechnicheskaya st.
195251 St. Petersburg Russia
ponmike92@gmail.com

Evgeny Pyshkin
University of Aizu
Software Engineering Lab
Tsuruga, Ikki-machi, Aizu-Wakamatsu
Fukushima, Japan 965-8580
pyshe@u-aizu.ac.jp

ABSTRACT

The paper is focused on an approach to computing structural syntactic similarity of mathematical equations presented in MathML. We examine a modification of a tree overlapping algorithm adopted to a purpose of describing mathematical equation similarity patterns.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; I.5.1 [**Pattern Recognition**]: Models—*structural*

General Terms

Algorithms, Human Factors

Keywords

Syntactic similarity, Mathematical equations, MathML, Tree overlapping

1. INTRODUCTION

Nowadays there is a few models of adopting natural language processing (NLP) algorithms related to syntax similarity to the specific notations used in mathematics. Mathematical equations (with their unique structural syntax with a big variety of semantically equivalent constructions) provide a non-trivial case for information retrieval [6]. Many reported implementations are focused on exact matching of mathematical constructions rather than on their similarity [5, 4]. Indeed, for a case of mathematical equations, syntactical similarity is defined rather fuzzy by using several structural syntactical similarity patterns. However, such a model would be very useful while developing searching and classification tools, especially used in education by math learners and tutors that would allow selecting suitable tasks to nail down a topic presented during a classroom session.

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

For example, there is an obvious use case which is accessing a set of relevant mathematical equations to be used for learner's training while doing preparation works for an examination. One more option is using such tools for searching an equation by its syntactical structure, the latter being often easier to recall comparing to exact mathematical formulas.

For the reason that most structural notations used for mathematical expression representation are in fact based on directed graphs, syntactic similarity can be defined by using tree structural similarity. Specifically, in this work we use the expressions uniformly presented in MathML[1] which is one of widely used structural XML based mathematical notations. In turn, if we use better structural forms to represent a math equation, we can expect more efficient and accurate retrieval in contrast to the not rare case of using image based equation representation on many web sites. At the same time we accept a possible criticism that not an every mathematical expression retrieval difficulty could be addressed under limitations of MathML representability.

2. STRUCTURAL SIMILARITY OF MATHEMATICAL EQUATIONS

In [1] similarity of two trees is defined on the base of recursive examination of their subtrees. In [3] the following mathematical expressions similarity patterns are defined:

Mathematical equivalence: Equations E_1 and E_2 are mathematically equivalent if they are semantically the same (but not obligatorily syntactically the same, for example $\frac{d(sin(x))}{dx}$ and $(sin(x))'$, $sin^2(x) + cos^2(x)$ and 1.

Identity: E_1 and E_2 are identical if they are exactly the same.

Syntactical identity: E_1 and E_2 are syntactically identical if they are identical after normalization (dealing with variable names and numeric values). For example $sin(a)$ and $sin(b)$, $\frac{1}{sin(x)}$ and $\frac{5}{sin(x)}$.

Expression n–similarity: Normalized equations E_1 and E_2 are n-similar if similarity (in a certain sense) $sim(E_1, E_2) \geqslant n$, n being a parametric value determining the threshold. There are two specific cases of n–similarity which are particularly important for our work:

1. **Subexpression n–similarity**: There is a subexpression n–similarity for E_1 and E_2, if they are n–similar and the corresponding trees both contain the common

[1]MathML – Mathematical Markup Language

subtree which in turn contains all the terminal nodes of both trees. Figure 1 shows an example for the case of $sin(x)^2$ and $\frac{sin(x)}{2}$.

2. **Structural n–similarity**: E_1 and E_2 are structurally n–similar if they are n–similar (in a common sense) and there is a common part in both trees rooted at root nodes of the compared trees with the production rules being the same for all the nodes in this part. Figure 2 illustrates this case for the equations $x + \sqrt{sin(a)}$ and $x + \sqrt{2b}$.

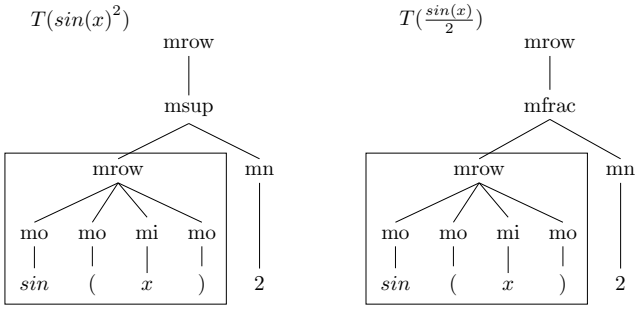


Figure 1: Equations $sin(x)^2$ and $\frac{sin(x)}{2}$ are structurally n–similar for any value of $n \geqslant \frac{18}{26}$

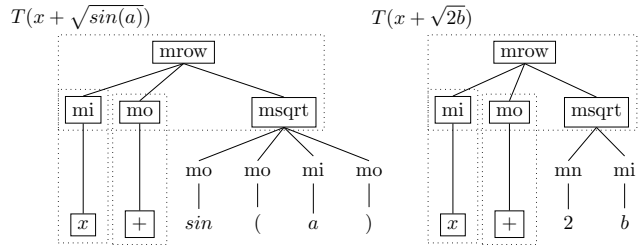


Figure 2: $x + \sqrt{sin(a)}$ and $x + \sqrt{2b}$ are structurally n–similar for any value of $n \geqslant \frac{12}{24}$

Note that in Figures 1 and 2 for n-values we use nodes ratio where a fraction's numerator corresponds to the number of the common nodes in both trees while its denominator corresponds to the number of all nodes in both trees.

2.1 Tree Overlapping Algorithm

A basic tree overlapping algorithm is described in [2] for a case of sentence similarity which is defined as follows. When putting an arbitrary node n_1 of a tree T_1 on a node n_2 of a tree T_2, there might be the same production rule overlapping in T_1 and T_2. Tree similarity is defined as a number of such overlapping production rules.

2.2 Modifying Tree Overlapping Algorithm for Math Equation Structural Similarity

In contrast to the base algorithm from [2] where tree terminals are naturally excluded, for a case of mathematical equations we propose to include terminal nodes as if they had the same production rules (*Relaxation 1*). Also we relax the strictness of the base algorithm and include the pairs of corresponding nodes which are in the same order among their siblings but do not have the same production rules for their child nodes (*Relaxation 2*). Below there is a formal definition of our modification.

Suppose $L(n_1, n_2)$ represents a set of overlapping node pairs when putting n_1 on n_2. It means that if $ch(n, i)$ is i-th child of node n then $L(n_1, n_2)$ is being generated by the following rules:

1. $(n_1, n_2) \in L(n_1, n_2)$
2. If $(m_1, m_2) \in L(n_1, n_2)$, then $(ch(m_1, i), ch(m_2, i)) \in L(n_1, n_2)$
3. $L(n_1, n_2)$ includes all the pairs generated recursively by the rule No. 2.

A number $N_{TO}(n_1, n_2)$ of production rules in question (according to the *Relaxation 1*) is defined as follows:

$$N_{TO}(n_1, n_2) = \left\{ (m_1, m_2) \left| \begin{array}{l} m_1 \in nodes(T_1) \\ \wedge\ m_2 \in nodes(T_2) \\ \wedge\ (m_1, m_2) \in L(n_1, n_2) \\ \wedge\ PR(m_1) = PR(m_2) \end{array} \right. \right\} \quad (1)$$

In equation 1 $nodes(T)$ is a set of nodes (including terminals) in a tree T, while $PR(n)$ is a production rule rooted at the node n.

Figure 3 shows an example of overlapping tree modification algorithm for $N_{TO}(d_1, d_2) = \{(d^1, d^2), (f^1, f^2), (g^1, g^2)\}$.

According to the *Relaxation 2*, suppose $P_{WPR}(n_1, n_2)$ is a set of nodes which is represented as a path from (n_1, n_2) to the top last pair of nodes being in the same order among their siblings. Suppose n_i and m_i are nodes of a tree T_i, $ch(n, i)$ is i-th child of node n. P_{WPR} is defined as follows:

1. $(n_1, n_2) \notin P_{WPR}$
2. If $PR(parent(n_1)) \neq PR(parent(n_2))$
   $\wedge\ ch(parent(n_1), i) = ch(parent(n_2), i)$
   $\wedge\ ch(parent(n_1), i) = n_1$
   $\wedge\ h(parent(n_2), i) = n_2$,
   $(parent(n_1), parent(n_2)) \in P_{WPR}$
3. $P_{WPR}(n_1, n_2)$ includes only pairs generated by applying rule No. 2.

Then the second component for an integral similarity measure can be defined bu using the above introduced P_{WPR} as follows:

$$P_{TO}(n_1, n_2) = \left\{ (m_1, m_2) \left| \begin{array}{l} (p_1, p_2) \in N_{TO}(n_1, n_2) \\ (m_1, m_2) \in P_{WPR}(p_1, p_2), \\ \qquad \text{if} \quad top(m_1, m_2) = (n_1, n_2) \end{array} \right. \right\} \quad (2)$$

In equation 2 $top(n_1, n_2)$ is the last pair in set $P_{WPR}(n_1, n_2)$:

$$top(n_1, n_2) = p_{last}(n_1, n_2), \qquad p_{last} \in P_{WPR} \quad (3)$$

Thus, for two nodes the resulting combined similarity measure is defined as follows:

$C_{TO}(n_1, n_2) = |N_{TO}(n_1, n_2)| + |P_{TO}(n_1, n_2)|$

For the whole trees, we get:

$$S_{TO}(T_1, T_2) = \max_{n_1 \in nodes(T_1), n_2 \in nodes(T_2)} C_{TO}(n_1, n_2) \quad (4)$$

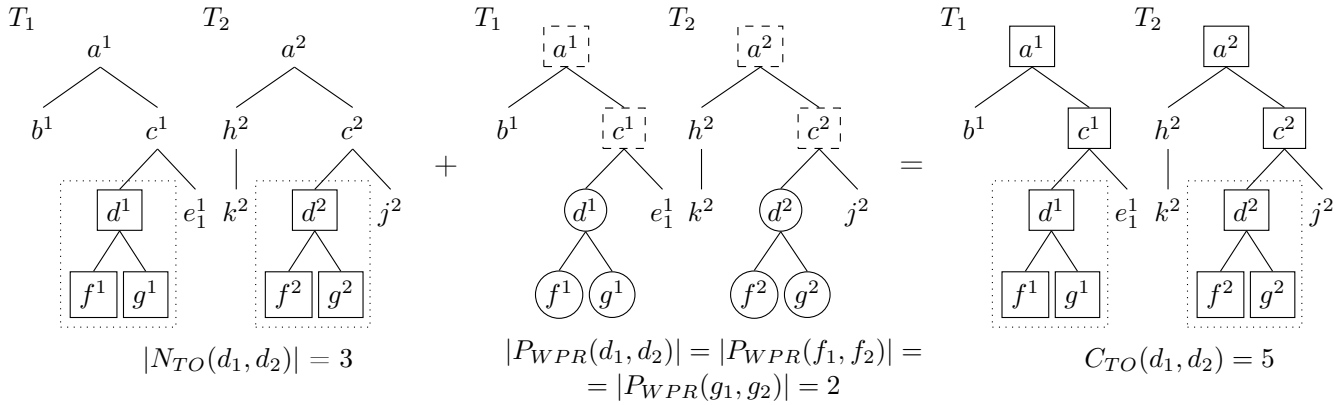


Figure 3: Modified Tree-Overlapping Algorithm: Example

2.3 Software Implementation

We developed a software prototype in order to arrange a series of experiments for our modification of the tree overlapping algorithm for a case of mathematical equations. Figure 4 gives a hint of how the application user interface is organized.

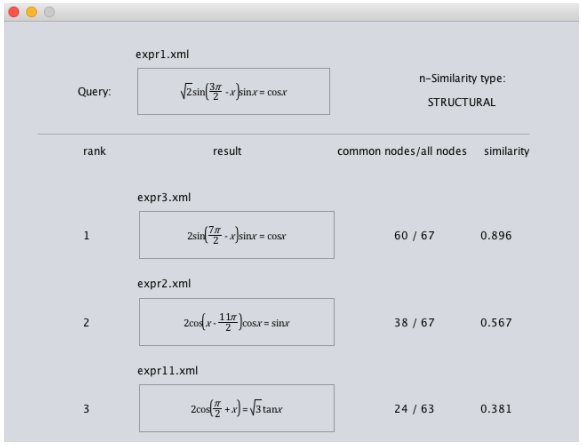


Figure 4: Structural similarity component: GUI

For displaying mathematical equations defined in MathML the library *net.sourceforge.jeuclid* is used.

3. EXPERIMENTS

There is a significant problem we faced while attempting to evaluate our modification algorithm. We discovered that, unlike to the NLP domain, there is no substantial corpus of mathematical equation syntactical similarity classes. So, for our rather preliminary analysis we selected a number of typical trigonometry problems from the set of tasks used in Russian National Exam on Mathematics. Then we involved several experts experienced in teaching mathematics. With their help we classified a selection of expressions in order to proceed with preliminary analysis of our approach.

3.1 Test Corpora

For our initial experiments we created a set of equations classified according their structural similarity (being limited by the paper size we skip here our tests for subexpression similarity). Table 1 lists the equations we used in our experiments.

Table 1: Expression Classification on Structural Similarity

No.	Expression	Class
1	$\sqrt{2}\sin(\frac{3\pi}{2} - x)\sin x = \cos x$	①
2	$2\cos(x - \frac{11\pi}{2})\cos x = \sin x$	
3	$2\sin(\frac{7\pi}{2} - x)\sin x = \cos x$	
4	$-\sqrt{2}\sin(-\frac{5\pi}{2} + x)\sin x = \cos x$	
5	$\cos 2x - 3\cos x + 2 = 0$	②
6	$\cos 2x + 3\sin x - 2 = 0$	
7	$3\cos 2x - 5\sin x + 1 = 0$	
8	$\cos 2x - 5\sqrt{2}\cos x - 5 = 0$	
9	$\cos(\frac{\pi}{2} + 2x) = \sqrt{2}\sin x$	③
10	$\cos 2x = \sin(x + \frac{\pi}{2})$	
11	$2\cos(\frac{\pi}{2} + x) = \sqrt{3}\tan x$	
12	$2\sin^4 x + 3\cos 2x + 1 = 0$	④
13	$4\sin^4 2x + 3\cos 4x - 1 = 0$	
14	$4\cos^4 x - 4\cos^2 x + 1 = 0$	
15	$(2\cos x + 1)(\sqrt{-\sin x} - 1) = 0$	⑤
16	$(2\sin x - 1)(\sqrt{-\cos x} + 1) = 0$	
17	$\sqrt{\cos^2 x - \sin^2 x}(\tan 2x - 1) = 0$	
18	$\cos^2 x - \frac{1}{2}\sin 2x + \cos x = \sin x$	⑥
19	$\frac{1}{2}\sin 2x + sin^2 x - \sin x = \cos x$	
20	$\tan x + \cos(\frac{3\pi}{2} - 2x) = 0$	⑦
21	$\cos x + \cos(\frac{\pi}{2} + 2x) = 0$	
22	$\frac{2\sin^2 x - \sin x}{2\cos x - \sqrt{3}} = 0$	⑧
23	$\frac{2\sin^2 x - \sin x}{2\cos x + \sqrt{3}} = 0$	

3.2 Tests

Though a corpus presented in Table 1 isn't representative enough, it allows us to have some preliminary similarity precision estimation. Let us note that the preliminary experiments described in this work serves us as a prove-of-concept example for investigating further necessary improvements of the developed algorithm. In the future tests a standard cross-fold validation procedure will be required in order to get trustworthy precision evaluation results.

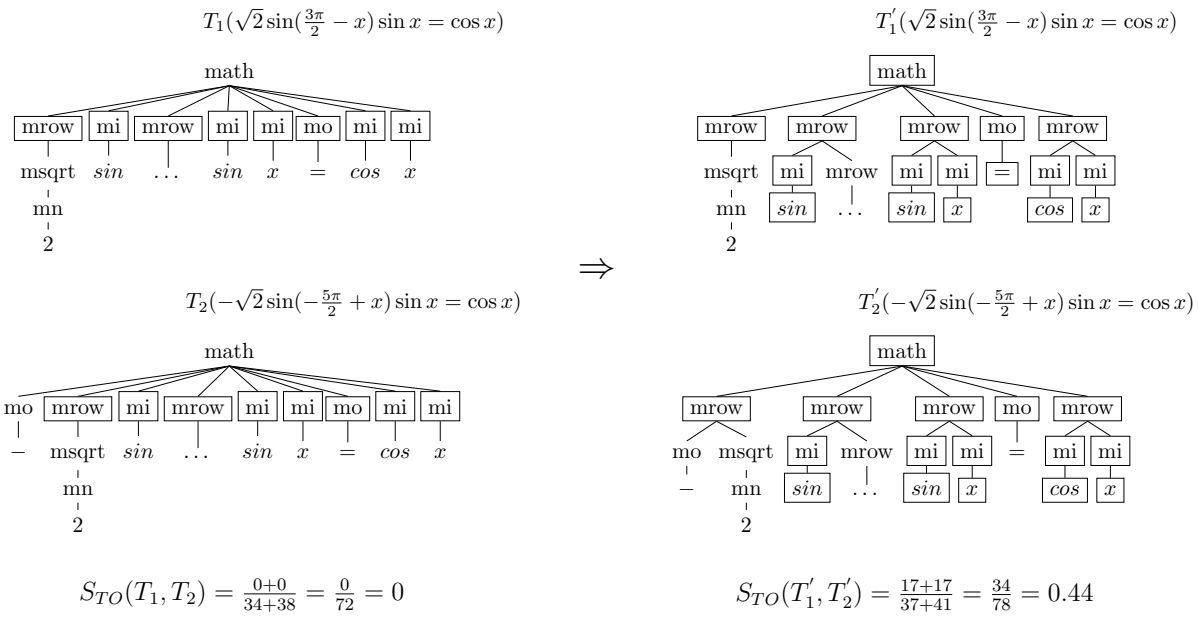


Figure 5: Tree structure normalization to avoid a false negative case

3.3 Analysis

In Table 2 we listed 5 expressions with the best scores for the query expression $\sqrt{2}\sin(\frac{3\pi}{2}-x)\sin x=\cos x$ (belonging to the class 1).

Table 2: Query: $\sqrt{2}\sin(\frac{3\pi}{2}-x)\sin x=\cos x$

Compared expressions	Nodes ratio	Similarity
$2\sin(\frac{7\pi}{2}-x)\sin x=\cos x$	60/67	0.896
$2\cos(x-\frac{11\pi}{2})\cos x=\sin x$	40/67	0.597
$2\cos(\frac{\pi}{2}+x)=\sqrt{3}\tan x$	24/63	0.381
$3\cos 2x-5\sin x+1=0$	12/60	0.200
$\tan x+\cos(\frac{3\pi}{2}-2x)=0$	10/67	0.149

Two best scores are for the equations which also belong to the class 1, except the equation $-\sqrt{2}\sin(-\frac{5\pi}{2}+x)\sin x=\cos x$ (No. 4 in Table 1), not recognized as a similar expression. To explain this phenomenon we have to go back to MathML equation structure. As you can see from Figure 5 (left side), two compared equations (both belonging to the class 1 of our corpus) have rather similar structure (at least, from the human point of view). However, their tree roots have different number of child nodes, hence their production rules are (formally) different. It means that we have to enhance equation normalization factor (currently limited by only variable names and numerical values): in the above mentioned case the issue can be resolved by restructuring a tree based equation representation as Figure 5 (right side) shows: both trees in the right side are semantically equivalent to those from the left side. However, similarity score increases from 0 (in the "left" case) to 0.44 (in the "right" case).

4. CONCLUSION

In our study of mathematical equation similarity patterns we adopted a tree overlapping algorithm (used originally in NLP) for mathematical equation syntactical similarity. After arranging a set of experiments, we discovered that our modification fits well a selection of equations from college-level teaching practice. We examined some drawbacks and argued that in order to improve precision the further steps towards equation normalization are required.

5. REFERENCES

[1] R. Bod. Beyond grammar. *An Experienced-Based Theory of Language. CSLI Lecture Notes*, 88, 1998.

[2] I. Hiroshi, H. Keita, H. Taiichi, and T. Takenobu. Efficient sentence retrieval based on syntactic structure. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 399–406. Association for Computational Linguistics, 2006.

[3] S. Kamali and F. W. Tompa. Improving mathematics retrieval. *Towards a Digital Mathematics Library. Grand Bend, Ontario, Canada, July 8-9th, 2009*, pages 37–48, 2009.

[4] K. Sain, A. Dasgupta, and U. Garain. Emers: a tree matching–based performance evaluation of mathematical expression recognition systems. *International Journal on Document Analysis and Recognition (IJDAR)*, 14(1):75–85, 2011.

[5] K. Yokoi and A. Aizawa. An approach to similarity search for mathematical expressions using mathml. *Towards a Digital Mathematics Library. Grand Bend, Ontario, Canada, July 8-9th, 2009*, pages 27–35, 2009.

[6] Q. Zhang and A. Youssef. *An Approach to Math-Similarity Search*, pages 404–418. Springer International Publishing, Cham, 2014.

Evaluation of Diffracted Wavefields Below the Salt Stringer Using the Transmission-Propagation Operator Theory and TWSM Software Package

Nikolay Zyatkov
Novosibirsk State University
Novosibirsk, Russia
nikolay.zyatkov@gmail.com

Alena Ayzenberg
University of Bergen
Bergen, Norway
alena.ayzenberg@uib.no

Kamaldeen Omosanya
NTNU
Trondheim, Norway
kamaldeen.o.omosanya@ntnu.no

Alexey Romanenko
Novosibirsk State University
Novosibirsk, Russia
arom@ccfit.nsu.ru

Arkady Aizenberg
IPPG SB RAS
Novosibirsk, Russia
AizenbergAM@ipgg.sbras.ru

ABSTRACT

We consider a seismic salt stringer image. The interpretation in the shadow zone beneath the stringer has complications due to that the diffracted and transmitted wavefields destructively interfere causing poor image. For simulating the real image we evaluate seismic wavefields in the shadow zone by combining the transmission-propagation operator theory (TPOT) and the tip-wave superposition method (TWSM) which is a mid-frequency range approximation of TPOT. This mathematical model has a layer with two flat boundaries, one of which has a dense coin-shaped addition reminding an anhydrite disk. We propose analytical separation of the wavefield below the stringer by TPOT and its computation by the TWSM software package. We describe usage of GPU-cluster to accelerate modeling and give an estimated time of wavefields simulation for stringer model.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; G.1.0 [**Numerical analysis**]: Parallel algorithms

General Terms

Theory, Algorithms, Performance

Keywords

seismic modeling, HPC, GPU

1. INTRODUCTION

Seismic interpretation beneath salt stringers is often complicated due to the difference in the physical and acoustic properties of the salt/evaporate and the stringers. Intra-salt deformation has been the focus of the oil and gas industries, and the academia in the last two decades [6], [7], [4]. Salt floaters are important reservoir rocks and geological storage in NW Europe and in the South Oman Salt Basin [5]. Apart from their economic importance, overpressure associated with salt stringers is drilling hazards, which are often hard to predict. The diffracted and transmitted wavefields in the shadow destructively interfere causing poor seismic imaging. There are several heuristic methods that separate primary reflections, surface-related multiples and internal multiples from the total wavefield in acoustic and elastic media [8]. We suggest the analytical separation from the total wavefield in multiphysics (acoustic, elastic, porous, fluid-saturated, etc.) block-layered media for the first time.

In this paper, we use the transmission-propagation operator theory (TPOT) to describe the acoustic blanking or shadow zone beneath a salt stringer in order to mitigate drilling hazards and provide better imaging of the intra-salt reservoir rocks. For simulating the real image we evaluate wave fragments of the interference wavefields into the shadow zone by highly-optimized software package based on the tip-wave superposition method (TWSM) that is a mid-frequency approximation of TPOT. We show usage of GPU-cluster to accelerate modeling by the TWSM package and give an estimated time of wavefields simulation for stringer model.

2. SALT STRINGER IMAGE AND ITS MATHEMATICAL MODEL

We consider a salt stringer image (Figure 1). By black ellipse, we emphasized the stringer part, which causes shadow beneath it and leads to this poor image. We aim to separate the wavefield in the shadow zone for better 'illumination'. Additionally, we consider a mathematical model (Figure 2) representing a large-scale picture of the stringer on Figure 1. This model consists of Sediments 1 and 2, Halite 1 and 2 and a thin Anhydrite disk. The interfaces have coordinate $z = -3.3$ km, $z = -4.0$ km and $z = -4.6$ km, respectively. The thickness of Anhydrite disk is ~ 0.042 km in length with a radius of 0.5 km. Sediments 1 and 2 have ve-

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

locity of 2.5 km/sec and density of 2.5 g/cm^3, respectively. Halite 1 and 2 have velocity of 4.5 km/sec and density of 2.2 g/cm^3. Anhydrite disk has velocity of 6.5 km/sec and density of 2.9 g/cm^3. A source is located in Sediments 1 at point: ($x = 0$ km, $y = 0$ km, $z = 0$ km). The source radiates a spherical P-wave represented by the scalar potential satisfying the Helmholtz equation. The wavelet is given by formula $e^{-(2\tau)^2}\cos(2\pi\tau)$, where $\tau = t/T - 2$, $T = 0.032$ s and dominant frequency 38.25 Hz. A 20 receivers array is spread in Sediments 1 along the line: ($x = -1.0$ to 0.9 km with the step $\Delta x = 0.1$ km, $y = 0$ km, $z = 0$ km). We consider two wave codes: reflection from Anhydrite disk (C1) and reflection from the bottom salt (C2) (Figure 2).

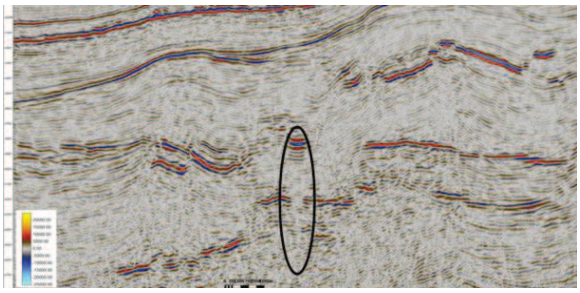


Figure 1: An example of a salt stringer, the Norwegian North Sea.

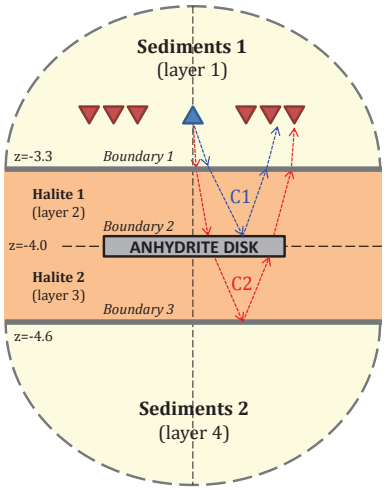


Figure 2: 3D mathematical stringer model.

3. TPOT&TWSM APPROACH IN SALT STRINGER MODEL

TPOT describes in explicitly analytic form propagation of the seismic wavefields in 3D block-layered geologic media by action of two operators: 1) transmission operator (reflection/refraction) $\mathbf{T}$ at curved interface, 2) propagation operator $\mathbf{P}$ inside block/layer [1], [2]. TPOT can evaluate not only the total wavefield but also structure of the feasible wave fragments in layered and macro-block media. Tip-wave superposition method (TWSM) being the mid-frequency range approximation of TPOT evaluates sequential action of composite operators $\mathbf{TP}$ in layers in according with the chosen wave code on the wave amplitude functions $\mathbf{a}$. Operators $\mathbf{TP}$ are approximated by layer matrices, and functions $\mathbf{a}$ – by layer vectors. Scalar elements of matrices and vectors are represented by analytical formulae [9].

TWSM evaluates the wave amplitude functions $\mathbf{a}^x$ by multiple multiplication of large scale matrices $\mathbf{P}$ and $\mathbf{T}$ and wave amplitude function $\mathbf{a}^{s_1}$ for source wavefield at interface $\mathbb{S}_1$. For stringer model wave amplitude function with wave code C1 at receiver can be represented as matrix-vector multiplication:

$$\mathbf{a}^x = \mathbf{P}^{xs_1} \cdot \mathbf{T}_{12} \cdot \mathbf{P}^{s_1 s_2} \cdot \mathbf{T}_{23} \cdot \mathbf{P}^{s_2 s_1} \cdot \mathbf{T}_{21} \cdot \mathbf{a}^{s_1} \qquad (1)$$

Wave amplitude function with wave code C2 at receiver can be represented as:

$$\mathbf{a}^x = \mathbf{P}^{xs_1} \cdot \mathbf{T}_{12} \cdot \mathbf{P}^{s_1 s_2} \cdot \mathbf{T}_{23} \cdot \mathbf{P}^{s_2 s_3} \cdot \mathbf{T}_{34} \cdot$$

$$\cdot \mathbf{P}^{s_3 s_2} \cdot \mathbf{T}_{32} \cdot \mathbf{P}^{s_2 s_1} \cdot \mathbf{T}_{21} \cdot \mathbf{a}^{s_1} \qquad (2)$$

Matrices $\mathbf{P}^{s_j s_i}$ describe propagation from elements of interface i to elements of interface j. Matrix $\mathbf{P}^{xs_1}$ describes propagation from elements of interface $\mathbb{S}_1$ to receiver x Matrices $\mathbf{T}_{kl}$ describe transmission from layer l in layer k. Matrices $\mathbf{T}_{23}$ in (1) and $\mathbf{T}_{34}$ in (2) describe reflections, other matrices $\mathbf{T}_{kl}$ describe refractions.

4. TWSM SOFTWARE PACKAGE

TWSM software package is highly-optimized realization of the TWSM for acoustic case adapted for GPU-cluster. Necessity to storage and process large scale matrices and vectors is a main problem of realization of the TWSM. Each interface of layered medium is triangulated by N triangles, each matrix of type 'interface - interface' is square and has dimension $N \times N$. Matrix of type 'interface - receiver' has dimension $M \times N$, where M is number of receivers. Wave amplitude vector of type 'source - interface' has dimension $N \times 1$. Each wave amplitude vector in the frequency domain demands to repeat matrix-vector multiplications (1) and (2) for each discrete frequency ω_k of some frequency array $\omega_1 ... \omega_K$. Acceleration of each matrix-vector multiplication is realized by GPU-cluster with help of scheme, shown at Figure 3. Each GPU accelerator processes multiplication of group of matrix strips corresponding to frequencies $\omega_1 ... \omega_K$ by group of wave amplitude vectors. Finally GPUs gather new transformed group of wave amplitude vectors via exchange by evaluated data. Since matrix dimension is $N \sim 10^5 - 10^6$, then TWSM software package can potentially keep any number of available GPUs but no more than N. Curve of scalability the TWSM software package is demonstrated at Figure 4.

5. TWSM COMPUTATION RESULTS FOR CODES C1 AND C2

For the computation of codes C1 and C2, we use formulae (1) and (2) with the transmission operators $\mathbf{T}$ and propagation operators $\mathbf{P}$. We approximate the transmission operators at interfaces 'Sediments1 - Halite1' and 'Halite2 - Sediments2' in these formulae by the plane-wave transmission (reflection/refraction) coefficients. The interface 'Halite1 - Halite2' has no contrast. For the thin Anhydrite disk with thickness Δz the transmission (reflection/refraction) coeffi-

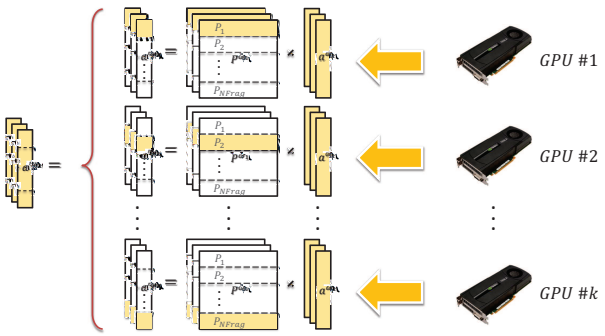


Figure 3: TWSM matrix-vector multiplication scheme adapted for GPU-cluster.

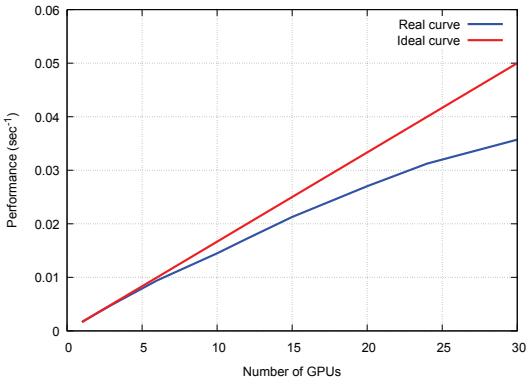


Figure 4: Scalability of TWSM software package.

cients of thin layer are [3]

$$R_{Disk} = T_{22}\frac{1-e^{i2\pi 2\Delta\bar{z}}}{1-e^{i2\pi 2\Delta\bar{z}}T_{22}^2}, T_{Disk} = \frac{(1-T_{22}^2)e^{i2\pi\Delta\bar{z}}}{1-e^{i2\pi 2\Delta\bar{z}}T_{22}^2}$$

In Anhydrite disk we have the dominant wave period $T_{dom} = \frac{1}{38.25Hz} = 0.026$ s, the dominant wavelength $\lambda_{dom} = 6.5$ km/s $*0.026$ s $= 0.169$ km, the thickness normalized by the wavelength $\bar{z} = \Delta z/\lambda_{dom} = 0.042/0.169 \approx 0.25$ and $e^{i2\pi 2\Delta\bar{z}} = e^{2\pi} = -1$. Halite - Anhydrite reflection at 0 degree is $T_{22} = (1/z_2 - 1/z_3)/(1/z_2 + 1/z_3) \approx 0.3113$, acoustic impedance for Halite is $z_2 = v_{P2} * \rho_2 = 4.5 * 2.2 = 9.9$ and acoustic impedance for Anhydrite is $z_3 = v_{P3} * \rho_3 = 6.5 * 2.9 = 18.85$. The reflection and transmission coefficients from Anhydrite disk are hence computed as follows

$$R_{Disk} \approx 0.567, T_{Disk} \approx 0.823e^{i\frac{\pi}{2}} \approx 0.823\frac{1+i}{\sqrt{2}}$$

Codes C1 (the upper curve) and C2 (the lower curve) are represented on Figure 5. Code C1 has stable and strong pulse shape, while code C2 has unstable and weak pulse shape. The change in the pulse shape is caused by the interference of the transmitted and diffracted wavefields. To better 'illuminate' the sub-stringer zone, we need to consider code C2 separately (Figure 6). The pulse of code C2 can be analyzed by removing Anhydrite disk (Figure 7). The difference between the seismograms on Figures 6 and 7 is illustrated on Figure 8, which gives understanding of the diffraction wavefield which we intend to remove from the whole modeling to get better 'illumination'.

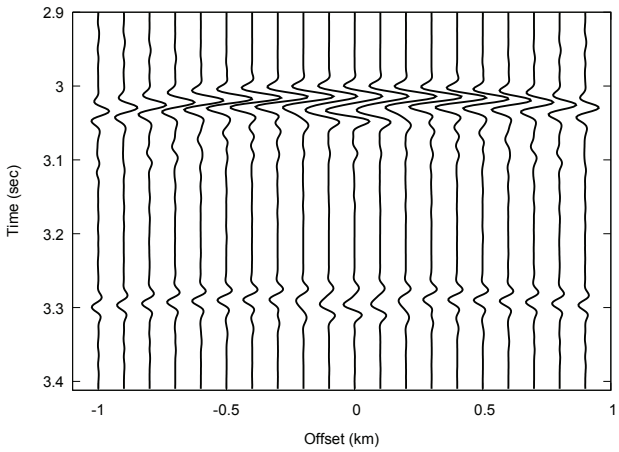


Figure 5: Codes C1 and C2.

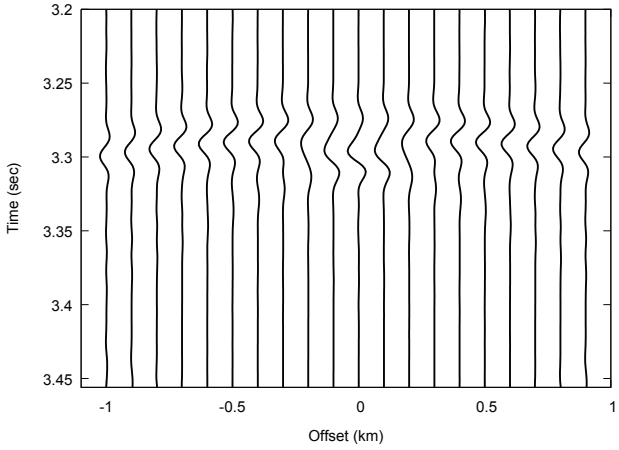


Figure 6: Separated code C2 from the common seismogram on Figure 5.

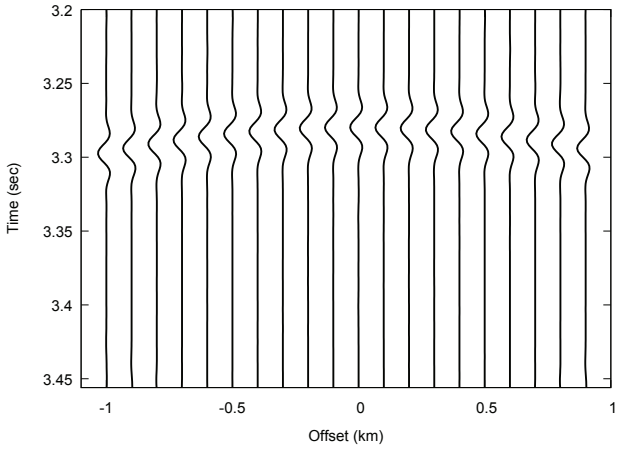


Figure 7: Code C2 with absence of Anhydrite disk.

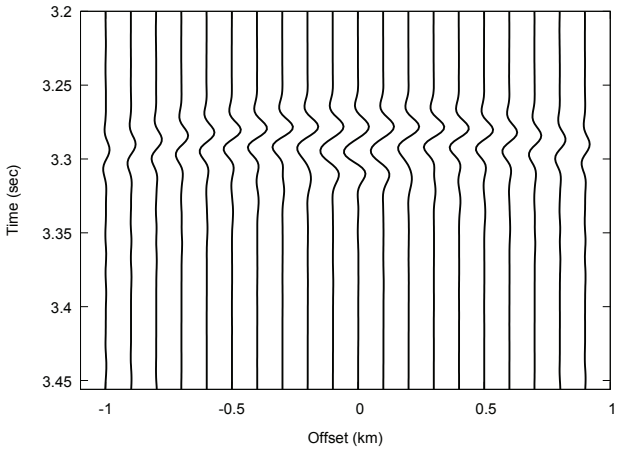


Figure 8: Difference of codes C2 with and without Anhydrite disk.

6. PERFORMANCE ANALYSIS OF TWSM FOR CODES C1 AND C2

Time of multiplication on 12 GPUs K layer matrices with dimension $N \times N$ and K wave amplitude vectors with dimension $N \times 1$ as function of number of triangles N at interface is shown in Table 1. We assume value $K = 128$ that corresponds to 128 discrete frequencies and time window 0.512 sec at seismograms. To decrease time of evaluation we apply rough approximation of the transmission operator with help of the plane-wave reflection/refraction coefficients. Let us estimate computational time of matrix-vector multiplication for wave codes C1 and C2 by formulae (1) and (2). In numerical tests each interface is triangulated by $N = 100000$ elements. We can neglect with computational time of evaluation of matrix 'interface - receivers' since it has dimension $M \times N$ when $M \ll N$. Using Table 1 we can obtain the calculation time for wave code C1 which will be $t = t_{P^{s_2}s_1} + t_{P^{s_1}s_2} = 38 + 38 = 76$ sec on 12 GPUs. Time calculation for wave code C2 will be $t = t_{P^{s_2}s_1} + t_{P^{s_3}s_2} + t_{P^{s_2}s_3} + t_{P^{s_1}s_2} = 38+38+38+38 = 152$ sec on 12 GPUs. For decreasing of computation time for wave codes C1 and C2 we can increase the number of involved GPUs in the TWSM software package.

Table 1: Time of filling and multiplication of group of square matrices and vectors as function of their dimension

Matrix dimensional (N)	Calculation time (sec)
10000	1
50000	9
100000	38
105000	83
200000	148
250000	229

7. CONCLUSIONS

In this paper, we consider a salt stringer image. Because the diffracted and transmitted wavefields destructively interfere, the image beneath the stringer is poor. To solve this problem we do analytical separation of the wavefield in the shadow zone beneath the stringer by TPOT and evaluate wavefields by highly-optimized TWSM software package adapted for GPU-cluster. This separation allows using only transmitted (or only diffracted) wavefield in evaluation and imaging, which will provide better 'illumination' of the shadow zone. The separated wavefield is obtained for a mathematical model simulating the stringer image. This mathematical model is a large scale sketch of the salt stringer. The separated wavefield is given in form of seismogram. The impact of stringer diffraction is presented on a separate seismogram.

8. ACKNOWLEDGMENTS

We are thankful to NTNU and the Research Centres Bergen and Trondheim of Statoil ASA. We thank Milana Ayzenberg, Jan Pajchel, Ola-Petter Munkvold, Alexey Stovas and Kees Wapenaar.

9. REFERENCES

[1] A. Aizenberg and A. Ayzenberg. Feasible fundamental solution of the multiphysics wave equation in inhomogeneous domain of complex shape. *Wave Motion*, 53:66–79, March 2015.

[2] A. Aizenberg, N. Zyatkov, A. Ayzenberg, and E. Rakshaeva. New concepts of the transmission-propagation operator theory in seismic diffraction modeling and interpretation. *76th EAGE Conference and Exhibition*, pages We–P06–07, June 2014.

[3] L. Brekhovskikh. *Waves in Layered Media*. New York: Academic Press, 1960.

[4] H. V. Gent, S. Back, J. Urai, and P. Kukla. Small-scale faulting in the upper cretaceous of the groningen block (the netherlands): 3d seismic interpretation, fault plane analysis and regional paleostress. *Journal of Structural Geology*, 32:537–553, April 2010.

[5] H. V. Gent, J. Urai, and M. de Keijzer. The internal geometry of salt structures - a first look using 3d seismic data from the zechstein of the netherlands. *Journal of Structural Geology*, 33(3):292–311, March 2011.

[6] Z. Gvirtzman, M. Reshef, O. Buch-Leviatan, and Z. Ben-Avraham. Intense salt deformation in the levant basin in the middle of the messinian salinity crisis. *Earth and Planetary Science Letters*, 379:108–119, October 2013.

[7] C.-L. Jackson, M. Jackson, M. Hudec, and C. Rodriguez. Enigmatic structures within salt walls of the santos basin - part 1: Geometry and kinematics from 3d seismic reflection and well data. *Journal of Structural Geology*, 75:135–162, June 2015.

[8] Y. Liu, H. Hu, X.-B. Xie, Y. Zheng, and P. Li. Reverse time migration of internal multiples for subsalt imaging. *Geophysics*, 80(5):S175–S185, September 2015.

[9] N. Zyatkov, A. Ayzenberg, A. Aizenberg, and A. Romanenko. Highly-optimized twsm algorithm for modeling cascade diffraction in terms of propagation-absorption matrices. *75th EAGE Conference and Exhibition*, pages Th–P02–11, June 2013.

A Tool for Bulk Deployment of Similarly Configured Resources in VMware vSphere

Andrei Kuznetsov
St. Petersburg Software Center
Motorola Solutions Inc.
Sedova st., 12
St. Petersburg, Russia, 192019
andrei.kuznetsov@motorolasolutions.com

Artyom Antonenko
Institute of Computer Science and Technology
Peter the Great St. Petersburg Polytechnic University
29 Polytechnicheskaya st.
St. Petersburg, Russia, 195251
artonenko@gmail.com

ABSTRACT

In this paper we consider a problem of bulk deployment of similarly configured resources in given resource provider. This problem often arises during testing of distributed enterprise applications, cloud services or infrastructure monitoring software. Manual deployment of virtual infrastructure is error-prone and time consuming when the infrastructure to be deployed is considerably large (e.g. contains more than 100 resource instances). Special bulk deployment assistant tools are very helpful for deployment of a large infrastructure to a resource provider. As a result of our examination of major bulk deployment use cases, we present a software implementation and its user interface. At this moment the tool supports configuring a virtual infrastructure in VMware vSphere, however the approach can be adopted for other resource providers (e.g. AWS, MS Azure, OpenStack, etc.).

Categories and Subject Descriptors

K.6.2 [**Installation Management**]: Computing equipment management; D.2.5 [**Testing and Debugging**]: Testing tools

General Terms

Experimentation

Keywords

VMware vSphere, Bulk deployment, Virtual infrastructure

1. INTRODUCTION

Rapidly growing hardware virtualization technologies have enabled development of many different kinds of distributed applications, including clustered enterprise applications and cloud-based applications and services. Increasing capabilities of these applications bring on their implementation complexity and require intensive application testing. In many cases testing of distributed applications requires special tools allowing developers to simulate high load conditions.

Let's consider testing of cloud provider (e.g. OpenStack, AWS, MS Azure, OpenShift, etc.) as an example. At some point test engineer might need to deploy 1000 resource instances (e.g. virtual machines in IaaS[1] cloud or application servers in PaaS[2] cloud) in order to test that a provider does not crash, there are no memory leaks or similar non functional problems, especially in a case if the number of deployed resources is considerably large. Another verification issue is related to the question whether a cloud service is scalable against available hardware resources. If there had been a need for implementing such a test, an engineer would have been allocating the available (virtual) hardware resources (for instance, about 100 or even more virtual machines). Then the testing engineer would have made cloud broker using all the available resources by deploying 10000 resource instances into the cloud. It's obvious that deploying such amount of resources into the cloud has to be automated in order to minimize manual routine work (and therefore, to minimize risks of deployment errors).

In this paper we consider VMware vSphere as a resource provider. The tool we describe in this paper is intended to support a software testing process in a company developing software for monitoring an infrastructure deployed in VMware vSphere. Let us note that the monitoring software is out of scope of this paper. In order to test this monitoring application it is required to deploy a homogeneous hierarchy of resources (which includes both resource pools and virtual machines) where each node of the hierarchy has a number of predefined parameters, while total number of nodes being more than 50. In the following sections we give overview of a resource hierarchy in VMware vSphere; we explain major use cases supported by our solutions and we investigate the issue of how this tool increases user performance and what are its major particularities against existing tools like *Veeam Backup & Replication*.

2. PROBLEM OF BULK DEPLOYMENT IN VMWARE VSPHERE

Virtual infrastructure deployment in VMware vSphere includes virtual machines and resource pools creation and configuration. Main settings of virtual machine are the follow-

[1]Infrastructure-as-a-Service
[2]Platform-as-a-Service

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

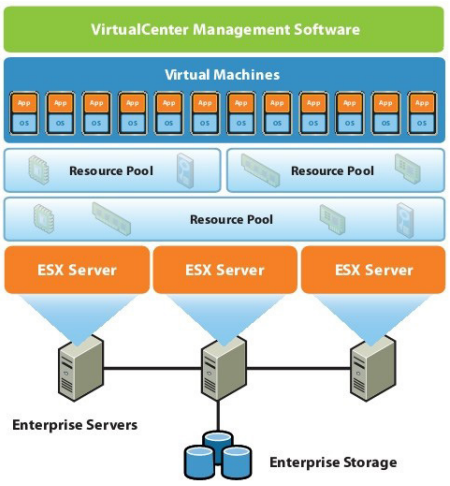


Figure 1: Resources hierarchy in VMware vSphere.[4]

ing: machine name, number of CPUs, memory size, virtual hard drive types and size[6]. The resource pool has only its name as one configurable parameter. If we have a few virtual instances (e.g less than 10 instances) it is not hard to configure them manually, but in a case of deploying a bulk (which might contain over 100 instances), manual configuration is time and effort consuming and might provoke errors.

In the following subsections we explain elements of VMware vSphere virtual infrastructure and standard deployment tools available for end users.

2.1 Resources hierarchy in VMware vSphere

Figure 1 demonstrates typical deployment of *vCenter Server*. The bottom layers (Enterprise Servers and Enterprise Storage) represent hardware; the middle layer (ESX Server) represents hypervisor software (or hardware) managing virtual resources; everything above hypervisor layer are virtual resources[5]. As we can see from the figure, the structure of *vCenter Server*'s virtual environment is hierarchical. There are resource pools which may include other resource pools and virtual machines[4]. In this way our challenge is to deploy a multi-tier structure with certain degree of nesting.

2.2 Standard VMware vSphere clients

In *vCenter* there is no user interface: in order to create and manage a *vCenter* infrastructure a number of special tools are provided by *VMware* including the following:

- *vSphere* desktop client;
- *vSphere* Web client;
- *PowerCLI* toolkit.

Table 1 summarizes advantages and drawbacks of the above mentioned tools in a case of being used for bulk deployment. As we can see, deploying a bulk of virtual resources with similar configuration is possible by using *PowerCLI* toolkit tools invocations in a special shell script [3]. Despite the script can generally solve the deployment problem, there are some issues which we have to consider from viewpoint of deployment tools usability and their user-friendliness:

[4]`http://www.assyrus.it/media/immagini/vmware_virtual_infrastructure.jpg`

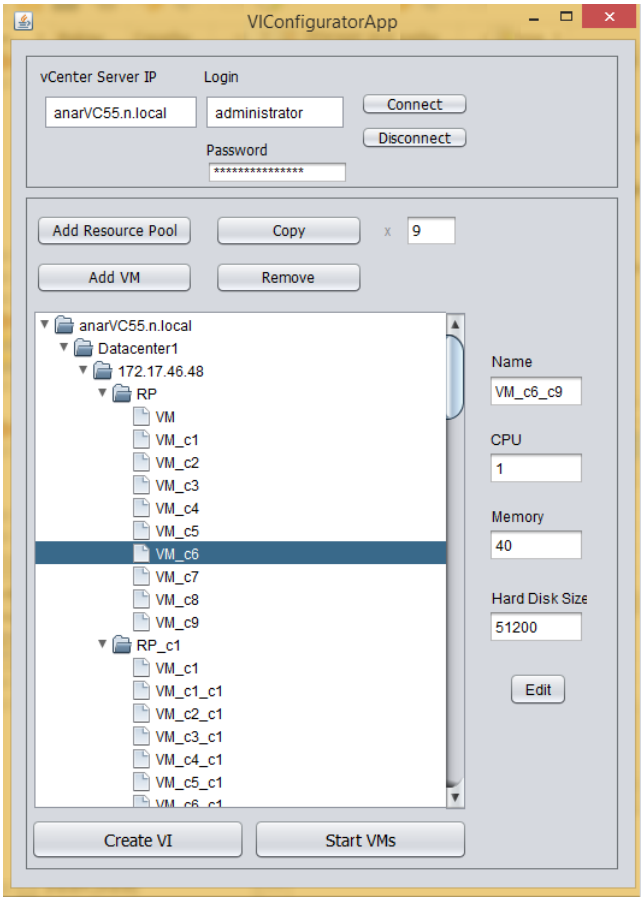


Figure 2: Main window of the developed application.

- A user has to specify a desired *hierarchical* infrastructure by using command line arguments or a configuration file. If we consider a single-level hierarchy consisting of virtual machines only, the user has to specify four arguments in order to configure main parameters of a virtual machine (virtual machine name, number of CPUs, memory size, hard disk size). One additional parameter is required to specify number of virtual machines. If we consider a case of more levels of hierarchy, script configuration might be much more complicated and a user will required to provide even more script parameters.
- A user is unable to see a hierarchy to be deployed before it is deployed (so as to check possible errors).
- Usually for an end user it is easier to learn a new tool which provides a graphical user interface (GUI) rather than a tool with no GUI[5].

[5]The best example of this statement in our practice is *vi* console text editor on Unix. It is usual that, once a user opens this editor, he or she asks about this text editor "How do I exit *vi*?". We have never heard about *notepad.exe* users on Windows wondering how to exit the *notepad.exe*.

Table 1: Standard VMware vSphere clients and their applicability for bulk deployment.

Tool	Advantages	Disadvantages
vSphere Desktop Client	There is a GUI	Bulk deployment requires too many manual routine user actions
vSphere Web Client	There is a GUI. No need to install.	Bulk deployment requires too many manual routine user actions
PowerCLI	Bulk deployment automation.	No GUI. All configuration parameters (e.g. VM configuration) are passed as command-line arguments (it makes script invocation complicated and increase the risks of user errors).

2.3 Deploying large infrastructure with vSphere Client

For demonstration purposes we consider a problem of deploying $N = 10$ virtual machines organized as $M = 10$ resource pools and its solution with vSphere Client (procedures for desktop and Web clients are the same). The following are the steps to be followed by a user:

1. Connect to vCenter Server by using the server name (IP-address or DNS name), user name and password. As soon as the tool is connected to the server, vCenter Server hierarchy view is loaded with using data obtained from the vCenter Server.
2. Select the host in the vCenter Server hierarchy view and click "New Resource Pool" button;
3. Specify a name and location for this resource pool;
4. Select the host in the vCenter Server hierarchy view and click "New Virtual Machine" button;
5. Select the configuration for the virtual machine(Typical or Custom);
6. Specify a name and location for this virtual machine;
7. Choose the resource pool you have created;
8. Select a destination storage for the virtual machine files;
9. Choose virtual machine version;
10. Specify the future guest operating system;
11. Select the number of virtual CPUs for the virtual machine;
12. Configure the virtual machine's memory size;
13. Configure network connections which will be used by the virtual machine;
14. Specify type of SCSI controller;
15. Select a virtual disk;
16. Specify the virtual disk size and provisioning policy;
17. Click the button "Finish";
18. Repeat steps 3-17 $N = 10$ times.
19. Repeat steps 2-18 $M = 10$ times.

Note that steps 3-17 (=15 steps) are executed $N \times M = 100$ times in total. Each step is 1 or 2 mouse clicks, so the user need to perform at least 1500 actions. Assuming that each action takes at least 1 second, the task cannot be completed faster than in 1500 seconds (25 minutes).

Despite that vSphere Client supports cloning of virtual machines (which allows the user to perform fewer actions), this option even more time consuming for the reason that clone operation takes 3 to 25 minutes to complete (depending on the virtual machine configuration). It's easy to calculate that populating 100 virtual machines via clone function will take at least 300 minutes (5 hours).

We believe that the infrastructure configuration time can be reduced significantly by improving "clone" function in the following ways:

- Clone function should be quick (i.e. no modifications should be applied immediately)
- Clone function should be available for resource pools (not only for virtual machines)

The following sections describe our implementation of the "clone" function and provide evaluation of the proposed tool implementing quick "copy" operation.

3. MAJOR USE CASES SUPPORTED BY OUR IMPLEMENTATION

This section explains major use cases supported by our application. Suppose that all virtual resources to be deployed have similar configuration. Despite such an assumptions is effectively a kind of constraint, this constraint allows us to introduce a "copy" operation applied to the resources in virtual infrastructure (resource pools and virtual machines). By using the "copy" operation the application allows users to create any number of copies of a certain resource (and, what is especially important, all of its subresources) in a couple of clicks. The procedure is demonstrated in the following subsections.

3.1 Deploying N Virtual Machines

The following are the steps to be performed by a user in order to deploy N (in our example we use $N = 100$) virtual machines into a *vCenter Server* infrastructure by using the developed tool:

1. Connect to *vCenter Server* by using the server name (IP-address or DNS name), user name and password. As soon as the tool is connected to the server, *vCenter Server* hierarchy view is loaded with using data obtained from the *vCenter Server* (see Figure 2).

2. Select the host in the *vCenter Server* hierarchy view and click "Add VM" button;
3. Select the created virtual machine;
4. (Optionally) Change the virtual machine parameters if necessary (name, number of CPUs, memory size, hard disk size);
5. Set number of copies to a desired value (99 in our example);
6. Click "Copy" button;
7. Click "Create VI" button in order to deploy the just configured hierarchy to a remote *vCenter Server*.

3.2 Deploying N Virtual Machines Organized as M Resource Pools

The following are the steps to be followed by a user in order to deploy N (in our example we use $N = 100$) virtual machines into a *vCenter Server* infrastructure under a condition that the VMs to be deployed are organized as M (we use $M = 10$) resource pools:

1. Connect to *vCenter Server* by using the server name (IP-address or DNS name), user name and password. As soon as the tool is connected to the server, *vCenter Server* hierarchy view is loaded with using data obtained from the *vCenter Server* (see Figure 2).
2. Select the host in *vCenter Server* hierarchy view and create a resource pool by clicking "Add Resource Pool" button.
3. (Optionally) Change the resource pool name;
4. Select the created resource pool and create one virtual machine;
5. (Optionally) Change the virtual machine parameters;
6. Select the virtual machine and create copies by specifying a desired value (9 in our example) in the corresponding edit box. Then click "Copy" button;
7. Select the resource pool and create $(M - 1)$ copies by specifying a desired value (9 in our example) in the corresponding edit box. Then click "Copy" button. Resource pool is copied with all its subresources (i.e. 10 virtual machines), such that in total the infrastructure contains $M = 10$ resource pools, where each resource pool contains 10 virtual machines;
8. Deploy the specified infrastructure by clicking "Create VI" button;

As a result of this scenario, a bulk of 100 virtual machines partitioned into 10 resource pools are created in the *vCenter Server* infrastructure.

4. IMPLEMENTATION AND EVALUATION

We implemented a bulk deployment assistant tool as Java 8 desktop application. For GUI implementation Java Swing library is used while for implementing communication with *vCenter Server* we use *VMware VI* (*vSphere*) Java SDK [1] library, version 5.5.

We emphasized two particular aspects while evaluating the developed application:

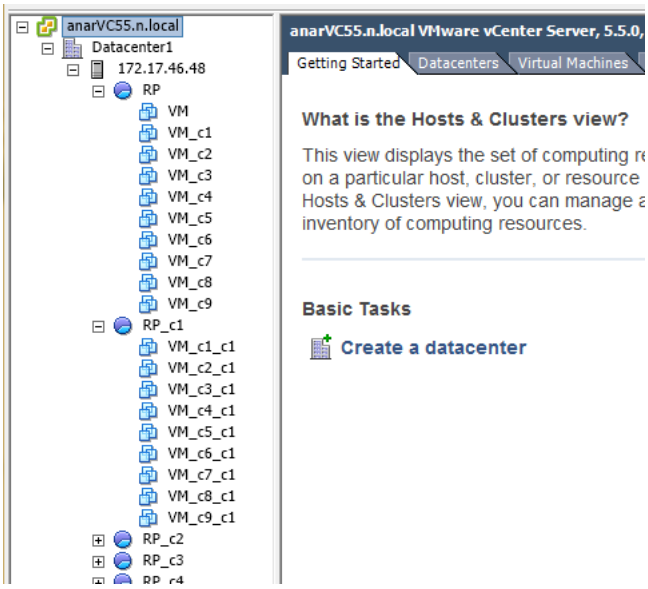


Figure 3: vSphere Client. View of the deployed infrastructure.

1. **Correctness**. This aspect is connected to checking whether an infrastructure created by a user using our application exactly matches a virtual infrastructure deployed into *vCenter Server*.
2. **User performance**. In order to go towards user efforts evaluation, we decided to compare the number of user actions (measured as mouse clicks) required to deploy a given infrastructure configuration into *vCenter Server* by using the standard *vSphere Client* against the case of using the developed deployment assistant tool.

In order to validate deployment correctness we use a manual scenario-based testing approach: the user configures an infrastructure with the help of our assistant tool, then vSphere Client is opened so as to allow viewing the actual infrastructure configuration just deployed. Figure 3 is an example of the actual infrastructure being observed with *vSphere Client* for a desired configuration presented in Figure 2. As you can see, the desired resource hierarchy matches the actual resource hierarchy, as well as each resource configuration within the hierarchy (e.g. virtual machine parameters) is as expected.

For user performance evaluation we executed a scenario (which is complex enough and which includes the resource pools and virtual machines configuration). This scenario was performed twice. In the first run we used the *vSphere* desktop client; in the second run we used the developed deployment assistant tool. The task was to deploy a virtual infrastructure with the following configuration: (root resource pool) $\xrightarrow[\times 10]{contains}$ (child resource pool) $\xrightarrow[\times 10]{contains}$ (virtual machine with a given configuration). In total there are 100 configured virtual machines and 11 configured resource pools. In the first run there were about 2500 mouse clicks, while in the second one there were about 30 mouse clicks. Such a tangible difference shows that the developed tool significantly simplifies bulk deployment of similarly configured

resources in *VMware vSphere* from the viewpoint of minimizing user efforts.

5. CONCLUSION

The tests we arranged for the developed application allow us to expect a significant reduction of efforts and user actions required in the time and effort consuming process for deploying virtual infrastructure in *VMware vSphere*. It is possible due to the implemented feature for massive copying of configured virtual machines and resource pools. The structure of virtual environment is represented in the form of a tree-like hierarchy which allows user to see the configured infrastructure.

Despite the described tool is a kind of highly specialized solution with limited functionality, it is appropriate for a concrete problem. We believe that our consideration is in a nice harmony with a Stroustrup's "clean code does one thing well"[2] – our tool does one thing well. The proposed application is not targeted to replace the existing virtual infrastructure configuration tools, it should be used side-by-side with the other useful tools.

After comparing our tool to a selection of existing *vSphere*-oriented tools we would agree that a problem of deploying a bulk can be solved by using replication features provided by some of them (for instance, *Veeam Backup & Replication* allows to deploy many instances of virtual resource from a backup image file). It is important to mention that the difference between our solution and the *Veeam Backup & Replication* tools (as well as other similar applications) is in defining the objectives. The primary objective of the *Veeam Backup & Replication* tools is to improve virtual infrastructure reliability and failure recovery time in a given virtual infrastructure. The primary objective of our tool is to configure a potentially huge number of resources with similar configurations within the context of a given resource provider, and *VMware vSphere* is the first one we targeted in our work[6].

[6]Support for other resource providers, including *AWS, OpenStack, OpenShift, MS Azure* etc. can also be provided in the future implementations.

6. REFERENCES

[1] S. Jin. *VMware VI and vSphere SDK: Managing the VMware Infrastructure and vSphere*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition, 2009.

[2] R. Martin. *Clean Code: A Handbook of Agile Software Craftsmanship*. Pearson Education, 2008.

[3] VMware, Inc. *VMware vSphere PowerCLI User's Guide*. VMware, Inc, 3401 Hillview Ave. Palo Alto, CA 94304, 1 edition, 2013. EN-001263-00, Available at `https://www.vmware.com/support/developer/PowerCLI/PowerCLI55/doc/vsp_powercli_55_usg.pdf`.

[4] VMware, Inc. *vSphere Resource Management*. VMware, Inc, 3401 Hillview Ave. Palo Alto, CA 94304, 2 edition, 2014. EN-001584-00, Available at `http://pubs.vmware.com/vsphere-55/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-552-resource-management-guide.pdf`.

[5] VMware, Inc. *vCenter Server and Host Management*. VMware, Inc, 3401 Hillview Ave. Palo Alto, CA 94304, 2 edition, 2016. EN-001520-04, Available at `http://pubs.vmware.com/vsphere-55/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-552-host-management-guide.pdf`.

[6] VMware, Inc. *vSphere Virtual Machine Administration*. VMware, Inc, 3401 Hillview Ave. Palo Alto, CA 94304, 2 edition, 2016. EN-001518-03, Available at `http://pubs.vmware.com/vsphere-55/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-552-virtual-machine-admin-guide.pdf`.

Day-Ahead Demand Management in Multi-Supplier Power Grid under Transmission Constraints

Ivan Popov
Control Methods and Robotics
TU Darmstadt, Germany
ipopov@rmr.tu-darmstadt.de

Alexander Krylatov
Saint Petersburg State University
Saint Petersburg, Russia
a.krylatov@spbu.ru

Victor Zakharov
Saint Petersburg State University
Saint Petersburg, Russia
v.zaharov@spbu.ru

ABSTRACT

Ever-increasing energy consumption and growing penetration of renewable energy sources stimulate the development of new power grid models and architectures. Since the decentralization of power grids raises the unreliability of power supply, it is crucial to switch to a production-oriented consumption in order to provide the stability of the grid. In this work, we describe a multi-supplier power grid model with day-ahead time span planning. We formulate and study a set of consumer cost minimization problems under flow distribution constraints. Finally, we consider an example illustrating the applicability of this model.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; I.2 [**Artificial intelligence**]: Problem Solving, Control Methods, and Search

General Terms

Theory

Keywords

Power Grid, Load Fow, Demand Management

1. INTRODUCTION

Traditionally, power grids have a central structure with a clear hierarchy. There are few power plants that produce and supply energy to a large area using transmission and distribution networks, and these power plants respond to a changing demand of consumers. However, due to the fast renewable energy development of recent decades, this situation is starting to change. New power grid architectures need to be created and studied in order to integrate smaller local renewable generators into the power grid while maintaining sustainability of the system.

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

One of the main challenges is providing balance between production and consumption in the network, especially taking into account the uncontrollable weather-dependent nature of main renewable energy sources (i.e., solar and wind energy). A possible solution is switching to production-oriented consumption, when consumers respond to changes in available generation capacities rather than producers to changing demands. This concept, also known as demand response management, includes different measures, but the goal is the same: to motivate consumers to change their strategies and to coordinate consumption with generators. Another issue is to maintain the transmission network and to avoid overloads in its links. This generally non-trivial problem becomes even more complicated for a decentralized system in the presence of multiple energy producers.

In this paper, we formulate and consider a multi-supplier power grid model, where consumers need to conclude bilateral contracts with suppliers over a day-ahead period of time divided in several time slots (e.g., 24 hours). The distribution of flows in the network deserves special attention, since it is crucial for preventing overloads and other disturbances in transmission lines. We describe consumers' costs as functions of their contract profiles, formulate a competitive game of consumers, and discuss possible schemes of demand response management for this model.

The topic of demand response was intensively studied in recent years. Demand management using pricing mechanisms for systems with a single generator and several competitive consumers are formulated in [10, 14]. Work [14] considers two-level piecewise linear cost functions, whereas in [10] functions are quadratic. Models with multiple generators and storage systems with quadratic costs are studied in [2, 3], where equilibria are found using variational inequalities. The ideas of cooperative game theory and coalition formation can also be applied for demand management (e.g., [9, 1]).

However, these works do not consider the flow distribution in the transmission network that depends on the topology of the network, whereas fulfilling lines capacities constraints is necessary for stability of the power grid. Some authors study networks with simple topologies: a set of parallel links [7], a network with a star-shaped structure [12], a network with a link for each producer-consumer pair [6]. In this work, we formulate a model with a general network topology and discuss challenges arising in this setting.

Flow distribution in electrical networks outside game theoretic scope is a well-studied topic, starting in the nineteenth century with formulation of Kirchhoff's current laws. The

problem of finding flow distribution was first formulated as a mathematical program in the middle of the twentieth century [5, 4]. It is known that methods of transportation cannot be applied to power flow distribution, since there are some critical differences between information and electricity, e.g., flows in electric networks cannot be routed directly [8]. However, one can note that Kirchhoff's laws and conditions of user equilibrium in the non-atomic routing setting are similar, and respective optimization problems take similar forms (see, e.g., Ch.2.6.3 in [11]).

The remainder of this work has the following structure. Section 2 describes the multi-supplier power grid model and the flow distribution in the transmission network. Section 3 formulates consumer cost minimization game and discusses the existence of equilibria. An example of demand response technics for the model is considered in Section 4, following by simulation results for a specific network. Finally, Section 5 concludes the paper and discusses future work.

2. MODEL DESCRIPTION

This section describes the structure of a power grid and discusses the power flow distribution in the grid. We also formulate a (generally non-linear) optimization problem for finding a power flow vector.

2.1 Network

A network is represented by a directed graph (V, A), where V is a set of nodes and A is a set of arcs. Let us enumerate nodes in V in the following manner: $V_Q = \{1, \dots, m\}$ is a set of m energy consumers, $V_P = \{m+1, \dots, m+n\}$ is a set of n producers, and $V_O = \{m+n+1, \dots, |V|\}$ is a set of all other nodes.

In this work, we consider a day-ahead planning period divided into H intervals. Each consumer concludes bilateral energy purchase contracts with several producers for each time interval. By e^h_{ij} we denote an amount of energy to be delivered from producer $j \in V_P$ to consumer $i \in V_Q$ during the time interval $h \in \mathbf{H} = \{1, 2, \dots, H\}$. We also use the following notation:

$$\mathbf{e}^h_i = (e^h_{i(m+1)}, \dots, e^h_{i(m+n)})^T \tag{1}$$

for a vector of i's contracts at a time interval h, and

$$\mathbf{E}_i = (\mathbf{e}^1_i, \dots, \mathbf{e}^H_i) \tag{2}$$

for a matrix of all i's contracts.

Consumers need to meet their energy demands, both total for the whole day and minimal for each time interval $h \in \mathbf{H}$. We denote the total demand of consumer i by $D_i \geq 0$, and the minimal demand of the same consumer for a time interval h by $d^{min}_i(h) \geq 0$. Therefore, we can write the demand constraints for $\mathbf{E}_i$:

$$\begin{aligned} \mathbf{1}^T_n \cdot \mathbf{e}^h_i &\geq d^{min}_i(h), \\ \mathbf{1}^T_n \cdot \mathbf{E}_i \cdot \mathbf{1}_H &= D_i, \end{aligned} \tag{3}$$

where $\mathbf{1}_k = (1, 1, \dots, 1)^T \in \mathbb{R}^k$.

Let us define energy balance b^h_k in a node $k \in V$ for a time interval $h \in \mathbf{H}$:

$$\begin{aligned} b^h_k &= -\mathbf{1}^T_n \cdot \mathbf{e}^h_k, && k \in V_Q, \\ b^h_k &= \sum_{i=1}^{m} e^h_{ik}, && k \in V_P, \\ b^h_k &= 0, && k \in V_O. \end{aligned} \tag{4}$$

This value reflects the amount of energy injected or withdrawn in a node during a specific time interval. It is negative for consumers and non-negative for producers, while we assume all other intermediate nodes to have zero energy balance.

Now we describe the flow distribution in the power grid for given energy balances.

2.2 Flow Distribution

Energy flows in a power grid are distributed according to Kirchhoff's laws, and we can find this distribution for a given set of energy balances and knowing parameters of grid links.

By $f^h_{kl} \geq 0$ we denote a flow in arc $(k, l) \in A$ at time interval h, and set $\mathbf{f}_h = \{f^h_{kl}, (k, l) \in A\}$ is a flow profile of all links at time h. We also define the following two subsets of V

$$\begin{aligned} W^{in}_k &= \{l \in V | (l, k) \in A\}, \\ W^{out}_k &= \{l \in V | (k, l) \in A\}, \end{aligned}$$

and the first Kirchhoff's law can be written as follows:

$$\sum_{l \in W^{out}_k} f^h_{kl} - \sum_{l \in W^{in}_k} f^h_{lk} = b^h_k, \quad \forall k \in V. \tag{5}$$

Let $\Theta_{kl}(f^h_{kl})$ be a voltage function for an arc $(k, l) \in A$, and by $\pi_k(h)$ denote an electric potential in a node $k \in V$ at time interval h. The second Kirchhoff's law takes the following form:

$$\pi_k(h) - \pi_l(h) = \Theta_{kl}(f^h_{kl}), \quad \forall (k, l) \in A. \tag{6}$$

The flow profile $\mathbf{f}_h$ can be found as a solution of a non-linear optimization problem (as in [11])

$$\underset{\mathbf{f}_h}{\text{minimize}} \quad \sum_{(k,l) \in A} \int_0^{f^h_{kl}} \Theta_{kl}(s) ds, \tag{7}$$

$$\text{subject to} \quad \sum_{l \in W^{out}_k} f^h_{kl} - \sum_{l \in W^{in}_k} f^h_{lk} = b^h_k, \forall k \in V \tag{8}$$

$$f_{kl} \geq 0, \quad \forall (k, l) \in A. \tag{9}$$

Since a set of contract vectors $\mathbf{E}^h = \{\mathbf{e}^h_1, \dots, \mathbf{e}^h_m\}$ defines the energy balances $\{b^h_k, k \in V\}$ which are parameters of the constraint set (8), we denote the solution of minimization problem (7)-(9) by $\mathbf{f}_h(\mathbf{E}^h)$. This mapping is generally non-linear, and contract changes of a single consumer affect the flow distribution in the whole grid.

3. GAME OF CONSUMERS

This section formulates and studies a consumer game as a model of interactions in the grid. First, we describe cost functions of consumers and formulate a game as a set of coupled cost minimization problems. In the second part of the section, the existence of Nash equilibria for the described game is discussed.

3.1 Consumer Cost Minimization

Each consumer tries to minimize their total costs over time span $\mathbf{H}$. These costs consist of two parts: generation costs and transmission costs. Generation costs can be assigned proportionally to the contracts between respective agents, while it is non-trivial to define the shares for use of transmission network.

More specifically, let $\alpha_j^h(b_j^h)$ denote a generation cost of a unit of energy at node $j \in V_P$ during time interval h. It is a function of total energy b_j^h to be generated at node j according to contracts with consumers $\mathbf{E}^h$. Hence, generation cost of consumer i during interval h can be determined in the following way:

$$G_i^h(\mathbf{E}^h) = \sum_{j=m+1}^{m+n} e_{ij}^h \cdot \alpha_j^h(b_j^h). \tag{10}$$

Transmission costs depend on the flow distribution $\mathbf{f}_h(\mathbf{E}^h)$. We define transmission cost for an arc $(k,l) \in A$ as a function $\beta_{kl}^h(f_{kl}^h)$ of the amount of flow using this arc. We call a set of functions $\Delta = \{\delta_{kl}^{i,h}(\mathbf{E}^h)\}$ a cost sharing rule, if it fulfills the following conditions:

$$\begin{aligned} &\delta_{kl}^{i,h}(\mathbf{E}^h) \geq 0, \quad \forall (k,l) \in A, i \in V_Q, h \in \mathbf{H}, \\ &\sum_{i=1}^{m} \delta_{kl}^{i,h}(\mathbf{E}^h) = 1, \qquad \forall (k,l) \in A, h \in \mathbf{H}. \end{aligned} \tag{11}$$

For a given cost sharing rule Δ the transmission cost of consumer i at interval h takes the form:

$$T_i^h(\mathbf{E}^h) = \sum_{(k,l)\in A} \delta_{kl}^{i,h}(\mathbf{E}^h) \cdot \beta_{kl}^h(f_{kl}^h(\mathbf{E}^h)). \tag{12}$$

Hence, the total cost of consumer i is

$$C_i(\mathbf{E}) = \sum_{h=1}^{H} \left(G_i^h(\mathbf{E}^h) + T_i^h(\mathbf{E}^h) \right), \tag{13}$$

where $\mathbf{E} = \{\mathbf{E}^1, \ldots, \mathbf{E}^H\}$ is a total profile of all consumer contracts over the whole time span $\mathbf{H}$, and where the calculation of each transmission cost $T_i^h(\mathbf{E}^h)$ requires solution of problem (7)-(9) for a respective time interval h.

We now formulate the game of consumers:

$$\underset{\mathbf{E}_i}{\text{minimize}} \quad C_i(\mathbf{E}), \quad 1 \leq i \leq m, \tag{14}$$

$$\text{subject to} \quad \mathbf{1}_n^T \cdot \mathbf{E}_i \cdot \mathbf{1}_H = D_i, \quad \forall i \in V_Q, \tag{15}$$

$$\mathbf{1}_n^T \cdot \mathbf{e}_i^h \geq d_i^{min}(h), \quad \forall i \in V_Q, \forall h \in \mathbf{H}, \tag{16}$$

$$e_{ij}^h \geq 0, \qquad \forall i \in V_Q, \forall j \in V_P, \forall h \in \mathbf{H}. \tag{17}$$

In this game, contract matrix $\mathbf{E}_i$ is a strategy of consumer i. We denote by $\mathbf{\Sigma}_i$ a set of all i's feasible strategies, i.e., a set of all matrices $\{\mathbf{E}_i\}$ fulfilling the conditions (15)–(17).

3.2 Existence of Nash Equilibria

The idea of Nash equilibrium proved to be the most appropriate solution concept for competitive games. A set of agents' strategies is in Nash equilibrium, if none of agents may reduce their total cost by unilaterally changing their strategy. In our model, a total profile $\mathbf{E}^*$ is in Nash equilibrium, if the following conditions are fulfilled:

$$C_i(\mathbf{E}^*) \leq C_i(\mathbf{E}_i, \mathbf{E}_{-i}^*), \forall \mathbf{E}_i \in \mathbf{\Sigma}_i, \tag{18}$$

where $\{\mathbf{E}_i, \mathbf{E}_{-i}^*\}$ is a total profile that differs from $\mathbf{E}^*$ only in component $\mathbf{E}_i$.

The existence of Nash equilibria in a consumer game strongly depends on the form of cost functions $\{\alpha_j^h(\cdot)\}$, $\{\beta_{kl}^h(\cdot)\}$ and the cost sharing rule Δ. Moreover, arguments of $\{\beta_{kl}^h(\cdot)\}$ are flows in the corresponding arcs, which are in turn components of a solution of non-linear optimization problem (7)-(9). Hence, establishing the fact of equilibrium's existence is a non-trivial task.

THEOREM 1. *Assume that a network contains no cycles, functions $\{\alpha_j^h(\cdot)\}$ are convex and increasing, functions $\{\beta_{kl}^h(\cdot)\}$ are convex, and transmission costs are shared according to rule (7)–(9). Then game (14)–(17) has a Nash equilibrium contract profile $\mathbf{E}^*$.*

PROOF. According to ([13]), an equilibrium exists for any n-person game with concave payoff functions. Since we consider cost functions rather than payoff functions, the same statement is true for games with convex cost functions. Therefore, we need to check whether a cost function $C_i(\mathbf{E}) = C_i(\mathbf{E}_1, \ldots, \mathbf{E}_m)$ is convex in $\mathbf{E}_i$ for each consumer $i \in V_Q$.

Function $C_i(\mathbf{E})$ consists of several summands:

$$C_i(\mathbf{E}) = \sum_{h=1}^{H} \left(G_i^h(\mathbf{E}^h) + T_i^h(\mathbf{E}^h) \right).$$

If we show that each summand in this sum is convex, convexity of the whole sum will be established as well. First, we study function $G_i^h(\mathbf{E}^h)$:

$$G_i^h(\mathbf{E}^h) = \sum_{j=m+1}^{m+n} e_{ij}^h \cdot \alpha_j^h(b_j^h). \tag{19}$$

When we fix the contract profiles of all consumers except i, function $\alpha_j^h(b_j^h + \lambda)$ remains convex and increasing, and function in (19) is convex as a product of two non-negative increasing convex functions.

Second, we rewrite function $T_i^h(\mathbf{E}^h)$ with fixed contract profiles of all consumers except i applying Proposition 3.1:

$$T_i^h(\mathbf{E}^h) = \sum_{(k,l)\in A} \delta_{kl}^{i,h}(\mathbf{E}^h) \cdot \beta_{kl}^h(f_{kl}^h(\mathbf{E}^h)). \tag{20}$$

The argument of $\beta_{kl}(\cdot)$ in (20) is a linear combination of $\{e_{ij}, j \in V_P\}$, components of consumer i's contract profile. Therefore, $\beta_{kl}^i(\mathbf{E}^h)$ remains convex in $\mathbf{E}_i^h$, as well as $T_i^h(\mathbf{E}^h)$.

The convexity of cost functions in respective arguments is established, that completes the proof. □

4. EXAMPLE

Consider a network with 7 nodes that is depicted in Figure 1. There are 3 consumers (red nodes), 3 producers (green nodes) and one intermediate node. Therefore, $V_Q = \{1,2,3\}$, $V_P = \{4,5,6\}$, and $V_O = \{7\}$.

All nodes are located in the same local area except for node 4 that depicts a conventional energy generator, e.g. a power plant. Hence, arc (4, 2) is longer than all other arcs, and transmission costs are higher for this arc.

Since there are no cycles in the network, we only need to check the first Kirchhoff's law (5). A flow on each arc is a linear combination of $\{e_{ij}^h\}$, $h = \{1,2,3,4\}$:

$$\hat{f}_{25}^h = e_{34}^h + e_{36}^h - e_{15}^h - e_{25}^h, \quad \hat{f}_{53}^h = e_{34}^h + e_{35}^h + e_{36}^h,$$

$$\hat{f}_{42}^h = e_{14}^h + e_{24}^h + e_{34}^h, \quad \hat{f}_{27}^h = e_{14}^h + e_{15}^h - e_{26}^h - e_{36}^h,$$

$$\hat{f}_{71}^h = e_{14}^h + e_{15}^h + e_{16}^h, \quad \hat{f}_{67}^h = e_{16}^h + e_{26}^h + e_{36}^h.$$

The direction of flow in arcs (2, 5) and (2, 7) may differ depending on the values $\{e_{ij}^h\}$. If $\hat{f}_{25}^h < 0$, we assign $\hat{f}_{25}^h = 0$ and $\hat{f}_{52}^h = -\hat{f}_{25}^h$. The same is true for $\hat{f}_{27}^h$.

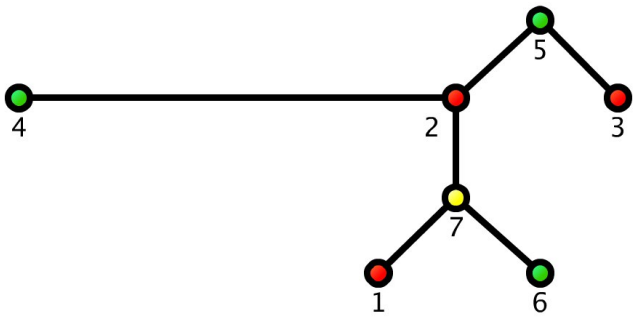


Figure 1: 7-node network with no cycles

Let us assume that functions $\{\alpha_j^h(\cdot)\}$ and $\{\beta_{kl}^h(\cdot)\}$ have the following form:

$$\begin{aligned} \alpha_j^h(x) &= \lambda_j^h \cdot x^{1+\epsilon} + \mu_j^h, \quad \forall j \in V_P, \\ \beta_{kl}^h(x) &= \lambda_{kl}^h \cdot x^{1+\zeta}, \quad \forall (k,l) \in A, \end{aligned} \tag{21}$$

where all coefficients are non-negative. We are ready now to solve the problem (14)–(17) with specific values of demands and coefficients in (21), and evaluate the total cost reduction. Actually, it is clear that the problem is a computationally difficult. Indeed, the presence of four time periods makes us to compare numerous combinations of different contracts. Thus, we are dealing with combinatorial optimization and the problem could be NP-hard. In future works we will investigate these questions carefully.

5. CONCLUSION

In this work, we have introduced new model for multi-supplier power grid under transmission constraints. Our model studies daily energy dynamics. The game of consumers was formulated and the existence result was established given specific properties of cost functions. There are several directions to improve and generalize the methods discussed in this work, and we name only few of them. First, real-world production and transmission costs, as well as voltage change functions, should be further studied in order to provide realistic representation of the network. Secondly, one can investigate a setting with dynamic network topology. Though power grid structures are relatively constant, there might be different applications of this model, e.g. for planning an optimal modification of a grid, or for maintaining the stability in a case of emergency such as blackouts.

6. REFERENCES

[1] M. Alam, S. D. Ramchurn, and A. Rogers. Cooperative energy exchange for the efficient use of energy and resources in remote communities. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pages 731–738. International Foundation for Autonomous Agents and Multiagent Systems, 2013.

[2] I. Atzeni, L. G. Ordóñez, G. Scutari, D. P. Palomar, and J. R. Fonollosa. Demand-side management via distributed energy generation and storage optimization. *Smart Grid, IEEE Transactions on*, 4(2):866–876, 2013.

[3] I. Atzeni, L. G. Ordóñez, G. Scutari, D. P. Palomar, and J. R. Fonollosa. Noncooperative and cooperative optimization of distributed energy generation and storage in the demand-side of the smart grid. *Signal Processing, IEEE Transactions on*, 61(10):2454–2472, 2013.

[4] J. B. Dennis. *Mathematical programming and electrical networks*. Technology Press of the Massachusetts Institute of Technology [Cambridge], 1959.

[5] R. Duffin. Nonlinear networks. iia. *Bulletin of the American Mathematical Society*, 53:963–971, 1947.

[6] J. S. Hong and M. Kim. Game-theory-based approach for energy routing in a smart grid network. *Journal of Computer Networks and Communications*, 2016, 2016.

[7] C. Ibars, M. Navarro, and L. Giupponi. Distributed demand management in smart grid with a congestion game. In *Smart grid communications (SmartGridComm), 2010 first IEEE international conference on*, pages 495–500. IEEE, 2010.

[8] J. Kok. *The PowerMatcher: Smart Coordination for the Smart Electricity Grid*. Amsterdam: Vrije Universiteit, 2013.

[9] R.-C. Mihailescu, M. Vasirani, and S. Ossowski. Dynamic coalition adaptation for efficient agent-based virtual power plants. In *Multiagent System Technologies*, pages 101–112. Springer, 2011.

[10] A.-H. Mohsenian-Rad, V. W. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia. Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid. *Smart Grid, IEEE Transactions on*, 1(3):320–331, 2010.

[11] M. Patriksson. *The traffic assignment problem: models and methods*. VSP, Utrecht, The Netherlands, 1994.

[12] I. Popov. Multi-supplier power grid framework based on multicommodity routing. In *International Energy and Sustainability Conference (IESC)*, pages 1–7. IEEE, 2015.

[13] J. B. Rosen. Existence and uniqueness of equilibrium points for concave n-person games. *Econometrica: Journal of the Econometric Society*, pages 520–534, 1965.

[14] A. Veit, Y. Xu, R. Zheng, N. Chakraborty, and K. P. Sycara. Multiagent coordination for energy consumption scheduling in consumer cooperatives. In *27th AAAI Conference on Artificial Intelligence*, pages 1362–1368. AAAI, 2013.

Air Cushion Vehicle Model for Educational Purposes

Ruslan A. Sevostyanov
Saint Petersburg University
Computer Applications and Systems Department
35 Universitetsky
Saint Petersburg, Russia, 198504
+79218833252
sevostyanov.ruslan@gmail.com

ABSTRACT
The paper introduces the model of the air cushion vehicle which is used to teach students control theory methods at the Saint-Petersburg State University. The model can operate either in the remote control mode or in the autopilot mode. Various tasks can be solved using this model: from the course stabilization to the dynamic positioning. Linear model and some experimental results are presented.

Categories and Subject Descriptors
I.2.9 **[Artificial Intelligence]**: Robotics – *autonomous vehicles*.

General Terms
Design, Experimentation.

Keywords
Control theory, education, marine systems, regulator, remote control, robotics.

1. INTRODUCTION
Nowadays we can observe the complication of the various vehicles. One of the brightest examples in the area of marine crafts is the class of air cushion vehicles (ACVs). The main feature of such crafts is the reduction of the water drag by creating the area of pressurized air between the water and the hull. For the crafts with the flexible skirt surrounding the air cushion this feature allows to operate not only in the water, but on the different surfaces: ground, ice, swamp, etc.

Due to the increasing requirements and restraints for the dynamics of all kinds of vehicles in different modes of operation there's a need in control theory specialists. Such specialists must know the whole process of the analysis of the plant dynamics and the synthesis of control law from the mathematical and computer model to the implementation of the control system. Thus, it's important to teach students not only the control theory methods, but also to provide a significant practice with various real-world vehicles.

For this purpose here at the Saint-Petersburg State University we've built the model of the ACV. It has 32-bit controller and several sensors, including inertial measuring unit and onboard webcam. The model can operate in two modes. The first one is the autopilot mode when there's a program executing on the controller. The second one is the remote control mode, either manually via the mobile application or automatically via remote computer.

The model can be used in various tasks, such as model identification, course stabilization and control optimization. Some complications can be added to the basic tasks, such as use of the dynamic observers, filters or introducing delays into the control system (in the remote mode). All these tasks require knowledge of applied mathematics and programming skills, giving a good representation of the process of designing real control system.

2. THE MODEL
2.1 Structure and actuators

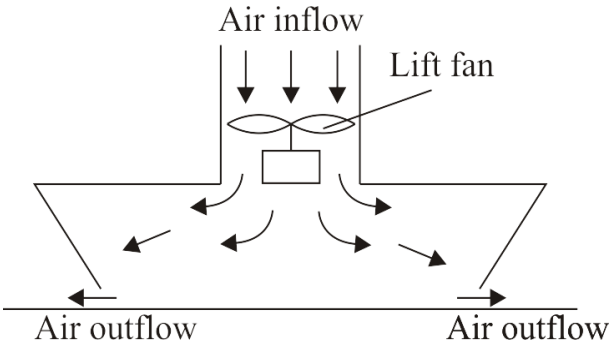


Figure 1. Plenum chamber air cushion.

The model represents a simple plenum chamber hovercraft where air directly fills the air cushion (see fig. 1). Although it is not the most efficient scheme for creating the air cushion it is the easiest one. There's one lifting fan and one thrusting fan (both with 1400 revolutions per minute per volt motors and individual voltage regulators). Thrusting fan is equipped with relay to change the direction of the motor. There are three synchronized air rudders with one servomotor to control the

*ICAIT'*16, Oct. 6–8, 2016, Aizu-Wakamatsu, Japan.

turn of the craft. The hull is made of extruded polystyrene (xps) and the skirt is made of artificial leather. Fan duct, rudders, lifting fan safety net and ultrasonic sensors' hulls are printed on the 3D printer. There are three 11.1V Li-Po accumulators: one for each fan's motor and one for the controller. The ACV model is presented on the fig. 2.

Figure 2. ACV model.

There's no cushion compartmentation (the air cushion is not divided into sections) so there're no restoring moments for the roll and pitch angles. Therefore the craft is quite sensible to the arrangement of all onboard components (especially accumulators). Non-zero roll angle produces lateral jet flow from the air cushion which affects the stability of the craft. All components must be placed quite carefully and fixed well to achieve satisfactory balance.

2.2 Controller

We're using TRIK controller [2] which is designed for using in various cyber-physical systems. It's based on the 375 MHz ARM9 processor and has 256 Mb of RAM memory. It also has additional discrete signal processor (DSP) which can, for example, process images from the camera without loading the main processor. The controller has onboard Bluetooth and Wi-Fi chip which can work either as network access point or as a client, so it is possible to communicate with TRIK remotely, for sending control commands or for streaming the camera video. There's also the peripheral processor which is capable of controlling all of the 19 general purpose digital ports and 4 motor ports. This processor allows to directly plug motors and servomotors to the controller without using additional drivers. There are also onboard gyroscope and accelerometer and one USB port.

There are several ways to program the controller depending on the students' programming skills. The easiest way is to use the open-source TRIK Studio IDE. There's a special visual language in which the program is structured as a sequence of visual blocks (see fig. 3). After creation the program can either be uploaded on TRIK via Wi-Fi or executed in the interpretation mode without uploading. It's also possible to use C++, C#, F#, Python, Java, JavaScript or QtScript languages. Such wide range of the available languages allows using this controller with minimal training time.

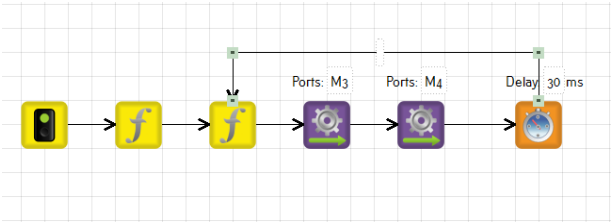


Figure 3. TRIK Studio program.

Another way is to build the remote system where the craft is used as an agent executing the control commands received from the remote computer and sending back its state's measurements. Here we can formulate two types of tasks: manual and autonomous remote control. In the first case students learn the basics of network communication. There's a convenient programming interface for this task made by TRIK team which allows creating such systems quite rapidly using the TCP sockets communication. In the second case the whole control system can be considered as a time-delay system and the corresponding problems can be formulated and solved.

2.3 Sensors

Although the controller has its own gyroscope and accelerometer it's very hard to get the stable measurements of the course angle without integration errors. So there's a need to use additional sensor such as magnetometer. We decided to use MPU-9250 inertial measurement unit which has 3-axis magnetometer, accelerometer and gyroscope. The main advantage of this chip is presence of the integrated digital motion processor (DMP) which receives data from chip's sensors and uses its own filters and algorithms for calibration and calculation of the stable and low-noise orientation angles. Of course it's also possible to get the raw data from each unit's sensor, so we can set and solve the filtering tasks also. MPU-9250 communicates with TRIK via the I2C interface.

There are three range sensors mounted on the front side of the ACV: two ultrasonic and one infrared. They allow solving tasks such as moving along the corridor and obstacle avoidance. In these tasks students can learn the basics of the proportional–integral–derivative (PID) regulators.

There's also the webcam which allows solving various computer vision tasks and to extend the use of the PID-regulators or to try more complicated methods (for example, visual servoing [1]).

3. PROJECTS AND FEEDBACK

During the last year we've been offering 3 projects to the students. The first project was to build the manual remote control system for the craft. The project was divided into 3 parts: developing the program in the TRIK Studio for receiving and performing the control commands on the onboard controller; developing the program for the desktop computer in the C++/Qt for sending control commands; developing the program for the onboard controller with the C++/Qt and replace the TRIK Studio program. Students that worked on this project said that it was quite interesting to build a system that actually allows you to control the real moving object and to feel what it's like to be the remote operator. And of course it was a great demonstration of the basics of the practical side of the network

communication and cross-compilation, because programs for the TRIK controller must be compiled on the desktop computer.

The two other projects are devoted to the control theory. The first of them was moving along the corridor. As mentioned in the previous section, there are three range sensors in front of the craft. The task was to move forward autonomously in the center of the corridor with some smooth turns and to stop or move backwards if there's and obstacle closer than 1 meter. This task introduces the students to the work with sensors and to the concept of the PID regulators. Most of the students working on this project said that at the first sight the task seemed much harder than it actually was and that after solving it they have a deeper understanding of what the concepts of feedback and regulator mean in control theory. Of course, the process of selection of the PID coefficients was not so easy and required a lot of experiments, but the concept itself was understood quite well and easily.

The last project was very close to the real problem for the different kinds of crafts. The task was to build an autopilot which must turn and keep the course angle on the desired level. In this project we're teaching the students the basics of the control law described in the next section. It requires knowledge of the linear model and could meet various requirements for the control quality. As we could conclude from the students' feedback, the main advantage of this task was not only the deeper understanding of the control law, but the connection between abstract mathematical models and real-world tasks, the idea that complex mathematical methods can be very useful in practical problems.

4. MULTIPURPOSE CONTROL LAW

Here we show the example of the educational task of the course stabilization in the simplest case. Let us consider the linear model of the craft. As stated in [4], linear models of the lateral motion of the real ACVs consist of drift, course and roll angles and of the angular velocity. But for our model it was enough to consider only angular velocity and course angle.

If we assume that the craft's moving with some constant longitudal velocity, zero angular and lateral velocities, zero rudders angle and without external disturbances, then the linear model of the course angle's dynamics can be expressed as follows:

$$\begin{aligned} \dot{\omega}_y &= a\omega_y + b\delta, \\ \dot{\varphi} &= \omega_y, \end{aligned} \tag{1}$$

where ω_y is the angular velocity, δ is the rudders angle and φ is the course angle. For our craft moving at the speed of 1.5 m/s coefficients in the right part of the linear system (1) have the following numerical values:

$$a = -0.147, b = 0.582.$$

In addition to system (1) we consider linear model of the rudders actuators:

$$\dot{\delta} = u, \tag{2}$$

where u is the control signal.

The system (1) – (2) is the linear time-invariant (LTI) system. If we assume that we can measure only the course angle, then the system (1) – (2) can be rewritten in the general matrix form:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \mathbf{A}\boldsymbol{x} + \mathbf{B}\delta, \\ \dot{\delta} &= u, \\ y &= \boldsymbol{C}\boldsymbol{x}, \end{aligned} \tag{3}$$

where y denotes the measurements and

$$\boldsymbol{x} = \begin{pmatrix} \omega_y \\ \varphi \end{pmatrix}, \mathbf{A} = \begin{pmatrix} a & 0 \\ 1 & 0 \end{pmatrix},$$
$$\mathbf{B} = b, \mathbf{C} = \begin{pmatrix} 0 & 1 \end{pmatrix}.$$

There exist various methods of stabilization of LTI-systems: pole assignment, linear quadratic regulators (LQR), PID regulators, etc. In our educational process we focus on the so-called multipurpose control laws [3] which take into account constant and periodical external disturbances such as wind or waves. In case of the wind the controller minimizes the course deflection and then makes the course angle converge back to the desired level. In case of the waves the controller can either provide accurate steering or minimize rudders deflections. One of the main advantages of such control laws is that the main task is divided into the smaller subtasks which are solved independently. It is also possible to switch off some parts of the controller for some computational economy, for example, on the calm water.

Multipurpose controller has the following structure:

$$\begin{aligned} \dot{\boldsymbol{z}} &= \mathbf{A}\boldsymbol{z} + \mathbf{B}\boldsymbol{\delta} + \mathbf{G}(\boldsymbol{y} - \mathbf{C}\boldsymbol{z}), \\ \xi &= F(p)(\boldsymbol{y} - \mathbf{C}\boldsymbol{z}),\ p = d/dt, \\ u &= \boldsymbol{\mu}\dot{\boldsymbol{z}} + \boldsymbol{\upsilon}\boldsymbol{y} + \xi. \end{aligned} \tag{4}$$

The first equation of the system (4) is the asymptotic observer which is used for estimation of the whole state of the system based on the measurements. Matrices **A**, **B** and **C** are from system (3). Vector $\boldsymbol{z}$ is the estimation of the state vector $\boldsymbol{x}$ and y denotes the measurements. Matrix **G** is selected so that the estimation converges to the state of the system.

The second equation of the (4) represents the dynamic corrector (or filter) which is responsible for dealing with periodical disturbances. Transfer matrix $F(p)$ is selected so that the effect of the disturbances is minimized and the control requirements are met.

Finally, the last equation of the (4) is the speed control law which provides stabilization of the system (3) with astatic property (i.e. stabilization to desired level even in the presence of constant disturbances) and minimizing the periodical disturbances. Coefficients $\boldsymbol{\mu}$ and $\boldsymbol{\upsilon}$ are selected in order to stabilize the system (3) with desired control requirements.

In our case asymptotic observer has the following form:

$$\begin{aligned} \dot{z}_1 &= az_1 + b\delta + g_1(\varphi - z_2), \\ \dot{z}_2 &= z_1 + g_2(\varphi - z_2), \end{aligned}$$

where $g_1 = 0.8412$ and $g_2 = 1.6378$.

Accordingly, the speed control law looks like this:

$$u = \mu_1 \dot{z}_1 + \mu_2 \dot{z}_2 + \upsilon\varphi,$$

where $\mu_1 = -3.1525$, $\mu_2 = -3.0483$ and $\upsilon = -1.8784$.

Fig. 4 demonstrates the craft's turn on 10 degrees without external disturbances.

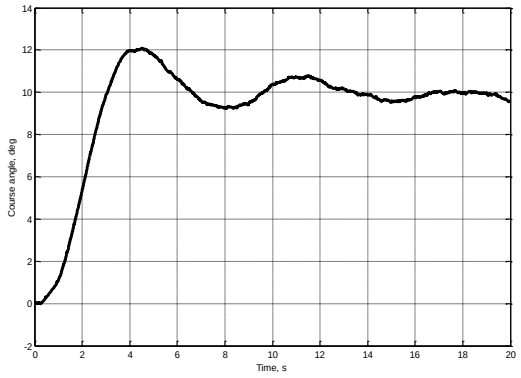


Figure 4. Course angle stabilization.

5. CONCLUSIONS

In this paper we presented the ACV model which we are using to teach student programming, applied mathematics and control theory methods. One year experience showed that it is much more interesting for students to work with real-world object and solve real tasks than to consider only computer models.

Of course there are still a lot of things that we wish to do. First of all, it is important to try cushion compartmentation to get the restoring moment and achieve zero roll and pitch angles. Another thing is to implement different technique of the air cushion formation: for example, momentum curtain, which should be more effective than simple plenum chamber cushion. It is also interesting to add second thrusting fan to make it possible to control the turn by the difference of thrusts.

So far we've been driving the craft only above the flat surfaces. Of course, it is very interesting to try motion above the water surface to get comparison to the dynamics above the usual flat surface as floor and to test the multipurpose control law in the case of water waves.

6. REFERENCES

[1] Hutchinsons, S., Hager G., Corke P. 1996. A Tutorial on Visual Servo Control. *IEEE Transactions on Robotics and Automation*, 12 (5), 651-670.

[2] Terekhov, A., Luchin, R., and Filippov, S. 2012. Educational Cybernetical Construction Set for Schools and Universities. *Advances in Control Education*, 9 (1), 430-435.

[3] Veremey, E. 2014. Dynamical Correction of Control Laws for Marine Ships' Accurate Steering. *Journal of Marine Science and Application*, 13 (2), 127-133.

[4] Yun, L., Bliault, A. 2000. *Theory and Design of Air Cushion Craft*. London, Arnolds.

Forecasting the Dynamics of Key Macroeconomic Indicators Based on Simulation

Lyudmila F. Vyunenko
St. Petersburg State University
62 Tchaikovsky st.
191123 St. Petersburg Russia
l.vyunenko@spbu.ru

Aleksey V. Vorontsovskiy
St. Petersburg State University
62 Tchaikovsky st.
191123 St. Petersburg Russia
a.vorontsovskii@spbu.ru

ABSTRACT
In this paper, we propose and explore a model for forecasting the development of the economy, formalized as the Cauchy problem for a system of stochastic differential equations taking into account "turning points". For the numerical solution of the corresponding computational problem, the Euler-Maruyama method is used. Stochastic simulation for different "what if" scenarios and several countries is performed using MATLAB.

Categories and Subject Descriptors
I.6.3. [**Simulation and modeling**]: Applications
J.4. [**Social and Behavioral Sciences**]: Economics

General Terms
Algorithms, Economics, Experimentation

Keywords
Macroeconomic indicators, dynamic models, Euler-Maruyama method, stochastic simulation, MATLAB

1. INTRODUCTION
The modern economy as it develops gains new properties and characteristics that make for a fresh look at the problems of forecasting economic dynamics including economic changes and crises. The main of them are connected with the expansion of globalization impact on the macroeconomic situation in separate countries and regions, the effect being significantly intensified due to the development of innovative processes. Models and methods designed nowadays to describe and forecast in macroeconomics, often rely heavily on the use of econometric models [3], [4]. This approach, based on preceding trends and patterns, requires large data sets containing macroeconomic indicators that are usually collected from different databases with respect to a long period of time. This greatly complicates the construction of reliable forecasts of economic development, taking into account the modern economic conditions. First, though econometric models use a large number of explanatory variables they are not free of missing essential macroeconomic variables or being based on the wrong variables. Secondly, in theory they rely on datasets where the time dimension is fairly long. But for many important data panels (e.g., for the Euro area or for the transition countries) the number of observations hardly exceeds 25-30.

An alternative is to consider dynamic models based on the idea that developments in the data are driven by only a few key factors [1], [6]. Provided these factors are identified the approach results in a small dimension model that may be used to produce "what if" scenarios and generate economic forecasts. Of particular importance for describing the development of the modern economy with the use of such models is the possibility of taking into account the random processes liable to influence the key factors, in particular, in the framework of stochastic dynamic general equilibrium models (SDGE).

Following [4] we make the difference between the terms "prediction" and "forecast" though they are interchangeable in many fields. The difference is not only etymological. We treat a prediction as a definitive and specific statement about the value of the indicator under consideration. Whereas a forecast is a probabilistic statement, usually over a long time scale.

We propose a dynamic model for forecasting that describes the changes in key macroeconomic indicators, taking into account only the current state of the economy. The model is described in Section 2. The discrete model underlying the simulation is shown in Section 3. The results of numerical experiments are illustrated in Section 4 with GDP dynamics for Finland and Japan. Section 5 concludes.

2. PROBLEM FORMULATION
Constructing the model, we took into account the following assumptions:
a) the conditions of the economic growth model for a closed economy are satisfied;
b) production function and technological changes are described in the form of an AK-model;
c) investment are instantly converted into capital (no lag).

Given these assumptions, the forecasting model has the form of the Cauchy problem for the system of stochastic differential equations (SDE) (see [5] for details):

*ICAIT'*16, Oct. 6–8, 2016, Aizu-Wakamatsu, Japan.

$$\begin{cases} dY(t) = (A(t) - \delta)K(t)dt + A(t)K(t)dW(t) \\ dK(t) = ((A(t) - \delta)K(t) - C(t))dt + K(t)dW(t) \\ dC(t) = A(t)K(t)dt - (1 - \delta)K(t)dt + A(t)K(t)dW(t) \end{cases} \quad (1)$$

with initial conditions $Y(0) = Y0$, $K(0) = K0$, $C(0) = C0$. $Y(t)$, $K(t)$ and $C(t)$ in (1) stand for gross output, capital stock and consumption at time t, $dY(t)$, $dK(t)$ and $dC(t)$ denote the increment of gross output, capital stock and consumption for the period $[t, t+dt]$, $A(t)$ denotes technological factor at t, δ is the capital depreciation rate. $dW(t)$ stands for the increment of the one-dimensional Wiener process (with the zero expectation and variance $\sigma_k^2 dt$).

3. DISCRETE APPROXIMATION

To construct a computable algorithm for solving the SDE system (1) we use the method for the approximate numerical solution of a stochastic SDE known as the Euler–Maruyama method [2]. It is a simple generalization of the Euler method for deterministic ordinary differential equations. The Euler–Maruyama approximation to the true solution is the Markov chain. The discrete approximation of (1) over $[0, T]$ and the time step Δ, provided the random variable dW is considered as discretized Brownian motion ($W(t)$ is specified at discrete t values), takes the form

$$\begin{cases} Y_{t+\Delta} = Y_t + (A_t - \delta_t)K_t\Delta + A_tK_t\sigma_k\sqrt{\Delta}\xi_t \\ K_{t+\Delta} = K_t + ((A_t - \delta_t)K_t - C_t)\Delta + K_t\sigma_k\sqrt{\Delta}\xi_t \\ C_{t+\Delta} = C_t + A_tK_t\Delta - (1 + \delta_t)K_t\Delta + A_tK_t\sigma_k\sqrt{\Delta}\xi_t \end{cases}, \quad (2)$$

where ξ_t are independent and identically distributed normal random variables with expected value zero and variance 1.

Note that for computational purposes the choice of the step size requires taking into account two considerations. First, the order of convergence for the Euler-Maruyama method is known to be low (strong order of convergence 0.5), so the results of calculations are inaccurate unless a small enough step size is used [5]. Secondly, the step should be chosen large enough to leave stochastic component meaningful.

4. SIMULATION

4.1 Software environment

When choosing the software environment for simulation, we were guided by the fact that MATLAB is an ideal one for generating and treating artificial data because of its

a) vectorizing core principle;
b) high-quality random number generators;
c) the opportunity to make experiments repeatable;
d) excellent graphics facilities;

Besides, an open architecture of MATLAB allows to modify any existing routine or to include new.

4.2 Simulation scenarios

We have performed a detailed three-part simulation study using data generation processes based on (2). In the first series of numerical experiments, we simulated the values of GDP, capital stock and consumption for 10 countries in different time periods and with different initial conditions. Master data for initialization the simulation were taken from statistical sources[1]. For parameters that are not available directly in the statistics, the procedure of model calibration has been applied, based on comparison of the simulation results with statistical data.

The results of the simulation showed that the model we proposed describes quite well the dynamics of key macroeconomic parameters for a short period of time (3-4 years), provided that there are no "turning points" - abrupt changes in the state of the economy.

Simulations of the second type were designed to account for the "turning points". To do this we used the technique of introduction of a dummy variable, a numeric stand-in for some qualitative fact (often used in time series analysis with regime switching). The appointment of dummy variable is to change abruptly the value of the capital depreciation rate δ at the particular time moment.

In the third part of the simulation study we performed "what if" scenarios for every of 10 countries: the baseline (inertial) one and the alternative one, taking into account the "shock" changes in the economy state at the "turning points". The baseline scenario assumes that changes occur to the extent that they could have been foreseen (no "turning points"). It corresponds to the constant value of depreciation rate δ during all the time period. The alternative case means short-term and medium-term effects of macroeconomic "shocks" caused by the external shocks (e.g. a fall in world oil prices, sharp fluctuations in world demand for exported goods and services, interest rates and exchange rates); the shocks due to the tariff policy of natural monopolies; fiscal shocks; the investment policy; reserve and monetary policy shocks; social policy shocks, etc. Determining the precise date of a "turning point" and the value of dummy variable that changes δ requires some time after the event has passed so simulation is the effective tool to for the analyzing "back" and determine the parameters that characterized previous periods or forecasting "forward" the dynamics of the economy state.

Countries we have chosen for the simulation experiments, varied considerably in the level and rate of economic development. The simulation results are illustrated below with the examples of GDP trajectories for two countries – Finland and Japan. We deliberately have chosen completely different examples with "turning points", caused by different economic reasons, to test the simulation model.

The greatest difficulties in applying the model are related to its calibration, which includes both the definition of the parameters that are not available in the statistics and the detection of "turning points" – those time points when the trend of the macroeconomic indicators development changes sharply. When carrying out simulation experiments for Finland and Japan we determined numerically the value of parameter δ in the equations (1), (2) and the value of the dummy variable responsible for its leap in "turning points", so as to ensure the conformity of the 50 % confidence domain for the mean of the

[1] http://www.be5.biz/makroekonomika/, http://www.ereport.ru/

simulated random process and the actual trajectories of the macroeconomic indicators. The exact date of each "turning point" was determined from the analysis of statistical data on macroeconomic indicators. Provided the necessary information is available, other approaches to the model calibration may be used, for example, based on statistical methods for the analysis of theoretical and calculated indicator trajectories compliance.

4.3 Macroeconomic indicators for Finland

Finland is traditionally characterized with a high level of income and well-being. The "turning points" in economy of Finland in 2004-2016 are connected with two events: the global downturn and the beginning of the war of sanctions and counter-sanctions that have dragged down the global output.

The Figure 1 shows the simulated GDP trajectories (thin dotted line), the average and the borders of the 50% confidence domain (bold dotted line), and statistic data (bold points).

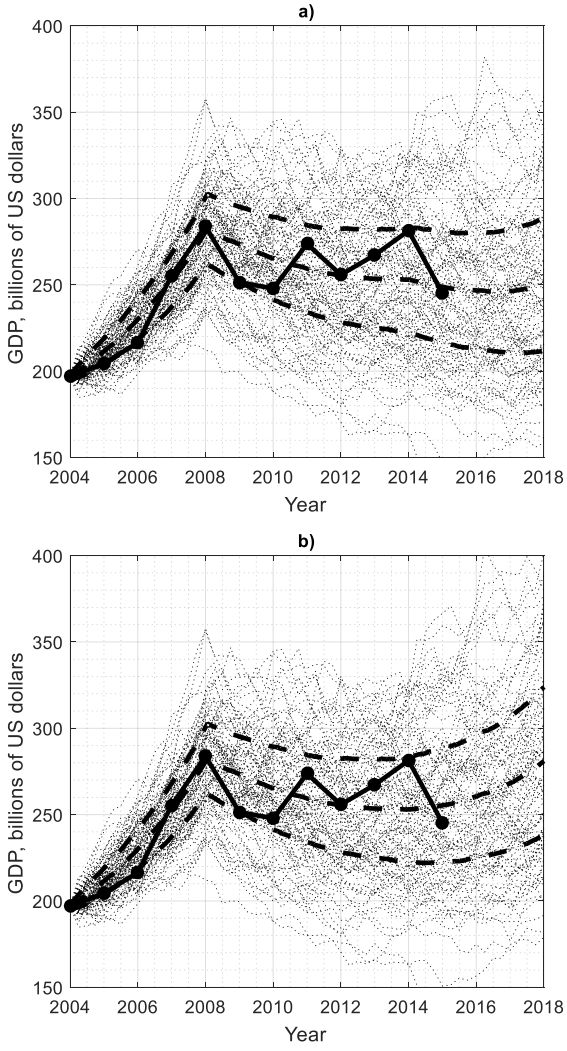


Figure 1. The GDP trajectories for the Finnish economy in 2004-2018: a) – two "turning points", b) - one "turning point".

The (a) subplot shows the dynamics of GDP trajectories simulated for a scenario that takes into account both turning points, the (b) subplot relates to the development of the process under the assumption that the second turning point is missing.

4.4 Macroeconomic indicators for Japan

The Japanese economy is one of the most developed economies in the world. The "turning points" for the economy of Japan in 2005-2016 are due to two events: the Global Economic Crisis that was not of great impact, and the unprecedented Triple Disaster 2011 that caused the immediate severe damage to Japan's economy.

The Figure 2 shows the simulated GDP trajectories (thin dotted line), the average and the borders of the 50% confidence domain (bold dotted line), and 2005-2015 statistic data (bold points). The (a) subplot corresponds to the "two turning points" scenario, (b) – to the "one turning point" variant.

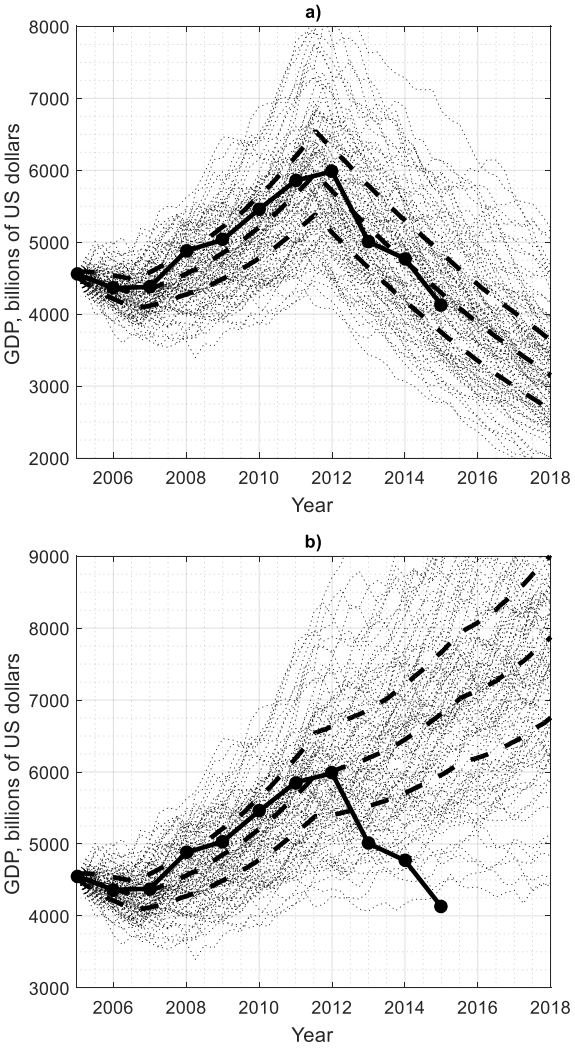


Figure 2. The GDP trajectories for the Japanese economy in 2005-2018: a) – two "turning points", b) - one "turning point".

5. CONCLUSIONS

The development of globalization and the increasing influence of uncertainty factors require methods to make forecasts based on the current state of the economy. In this paper, we proposed a computationally simple approach for forecasting the dynamics of key macroeconomic indicators. The forecasting model has the form of the Cauchy problem for the system of stochastic differential equations and its discrete approximation based on the algorithm of the Euler-Maruyama method. It allows arranging simulation study taking into account various scenarios including "turning points".

The advantages of the proposed approach are the possibility to quickly adjust the forecasts at the first sign of a changing macroeconomic trends, and the ability to solve the inverse problem, i.e., identify the model parameters that are not available in the statistics by comparing the simulation results and the statistical data.

The main problems encountered in the application of the approach, are connected with the model calibration, the justification of the economic information for setting initial conditions, determining the date of a "turning point" and the corresponding value of the dummy variable.

Further development of the forecasting model can be associated with the use of an improved algorithm of Euler-Maruyama, based on the use of more sophisticated approximations of stochastic differential equations.

6. REFERENCES

[1] Aivazian S., Brodsky B. *Macroeconometric modeling: modern trends, problems, an example of the econometric model of the Russian economy*. - J. Appl. Econometrics. 2 (2006), 85-111. Available at https://ideas.repec.org/a/ris/apltrx/0087.html (in Russian, accessed 07.07.2016).

[2] Desmond J., Higham. *An algorithmic introduction to numerical simulation of stochastic differential equations*. SIAM Review, 43, 3 (Sep. 2001), 525–546. Stable URL: http://www.jstor.org/stable/3649798.

[3] Dougherty C. *Introduction to Econometrics*. 4rd ed. Oxford University Press. 2011.

[4] Stock, J. H. and Watson, M. W. *Introduction to Econometric*s. Update, 3rd ed. Pearson Addison-Wesley. 2010.

[5] Vorontsovskiy, A. V., Vyunenko, L. F. *Constructing of economic development trajectories by approximating conditions of stochastic models of economic growth.* Vestnik SPbU. Ser. 5 (2014), 123-147. Available at http://vestnik.spbu.ru/html14/s05/s05v3/07.pdf (in Russian, accessed 07.07.2016).

[6] Wickens M. *Macroeconomic Theory: A Dynamic General Equilibrium Approach*. 2nd ed. Princeton University Press. 2011.

A Dual Formulation of The Traffic Assignment Problem for OD-matrix Estimation

Alexander Krylatov
Saint Petersburg State University
7/9 Universitetskaya nab.,
St. Petersburg, 199034 Russia
a.krylatov@spbu.ru

Anastasiia Shirokolobova
Saint Petersburg State University
7/9 Universitetskaya nab.,
St. Petersburg, 199034 Russia
st023662@student.spbu.ru

Victor Zakharov
Saint Petersburg State University
7/9 Universitetskaya nab.,
St. Petersburg, 199034 Russia
v.zaharov@spbu.ru

ABSTRACT

Congestion, accidents, greenhouse gas emission and others seem to become unsolvable problems for all levels of management in modern large cities worldwide. The increasing dynamics of motorization requires development of innovative methodological tools and technical devices to cope with problems emerging in the road networks. Primarily, control system for urban traffic area has to be created to support decision makers by processing a big volume of transportation data. The input for such a system is a volume of travel demand between origins and destinations — OD-matrix. The present work is devoted to the problem of OD-matrix estimation. The original technique of OD-matrix estimation is offered by virtue of plate scanning sensors location. Mathematically developed technique is based on a dual formulation of the traffic assignment problem (equal journey time by alternative routes between any OD-pair). Traffic demand between certain OD-pair is estimated due to journey time obtained from plate scanning sensors. Moreover, the functional relationship between traffic demand and journey time is obtained explicitly for the network of parallel routes with one OD-pair. Eventually, the developed method has been tested on the experimental data of the Saint-Petersburg road network.

Categories and Subject Descriptors

G.1.6 [**Mathematics of Computing**]: Numerical analysis—*optimization*

General Terms

Theory

Keywords

OD-matrix estimation, traffic assignment problem, duality theory

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

1. INTRODUCTION

OD-matrix estimation and reconstruction are urgent and complicated challenges, since road networks of modern cities are extremely large and intricate. In general, OD-matrix estimation and reconstruction are different problems: the first means to obtain approximate values, while the second means to obtain precise values of an actual traffic demand [1]. One of the first mathematical models for OD-matrix estimation was formulated in a form of bi-level program [2]. Despite numerous publications, this problem still attracts researchers from all over the world [3–8]. A detailed comparative analysis of the methods for trip matrix estimation was made in [4]. From a practical perspective, the most promising technique for trip matrix estimation is combination of data obtained both from plate scanning sensors and link-flow counts [5].

This paper is also devoted to the problem of OD-matrix estimation. We believe that a plate scanning sensor is the highly efficient engineering equipment. Indeed, due to link-flow counts one could obtain solely amount of vehicles on the link, while plate scanning allows estimating the average travel time between origin and destination by identification the vehicle in the origin and destination points. Since the travel time between an origin-destination pair is a Lagrange multiplier for a primal traffic assignment problem (TAP), it is the variable in a dual formulation of TAP. Therefore, we are able to formulate a new bi-level optimization program for OD-matrix estimation based on data from link-flow plate scanning sensors on congested networks.

The rest of this paper is organized as follows. In Section 2 the network of parallel routes with one OD-pair is investigated. The idea of OD-matrix estimation based on information about travel times between OD-pairs is clarified. Section 3 provides a dual formulation of the traffic assignment problem for a general topology network. Section 4 describes a bi-level optimization program for OD-matrix estimation on a congested network by virtue of plate scanning sensors. Section 5 is devoted to the experimental implementation of the developed approach to the Saint-Petersburg road network. Conclusions are given in Section 6.

2. THE NETWORK OF PARALLEL ROUTES

Let us consider a transportation network presented by a digraph with one OD-pair. Let us introduce the following notation: F is the traffic demand between OD-pair; f_i is the traffic flow on the route i, $i = \overline{1,n}$, $f = (f_1, \dots, f_n)$,

$\sum_{i=1}^{n} f_i = F$; $t_i(f_i) = a_i + b_i f_i$ is the travel time on congested arc i, $i = \overline{1,n}$. In the present work we model the travel time on the congested arc as the linear function.

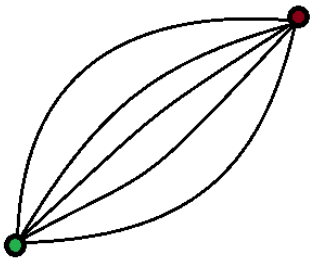


Figure 1: The road network of parallel routes with one OD-pair

Let us formulate the traffic assignment problem on the network of parallel routes as an optimization program [9, 10]:

$$z(f^*) = \min_f z(f) = \min_f \sum_{i=1}^{n} \int_0^{f_i} t_i(u)du, \tag{1}$$

with constraints

$$\sum_{i=1}^{n} f_i = F, \tag{2}$$

$$f_i \geqslant 0 \quad \forall i = \overline{1,n}. \tag{3}$$

Wardrop's first principle states that the journey times in all routes actually used are equal and less than those that would be experienced by a single vehicle on any unused route [10, 11]. The traffic flows that satisfy this principle are usually referred to as "user equilibrium" (UE) flows, since each user chooses the route that is the best. On the network of parallel routes UE is reached by such assignment $f^* = (f_1^*, \ldots, f_n^*)$ as:

$$\begin{cases} t_i(f_i^*) = t^* > 0 & \text{when } f_i^* > 0, \\ t_i(f_i^*) > t^* & \text{when } f_i^* = 0, \end{cases} \quad i = \overline{1,n}.$$

Thus, the mathematically formalized idea of UE (1)–(3) can be used in reconstruction of traffic assignment on the network between the origin-destination pair. However, if it is the travel time t^* between OD-pair that is known, we are able to reconstruct traffic demand F on the linear network of parallel routes.

Without loss of generality we assume that the routes are numbered as follows:

$$a_1 \leqslant \ldots \leqslant a_n.$$

THEOREM 1. *The traffic demand F for a linear network of parallel routes can be obtained explicitly:*

$$F = t^* \sum_{s=1}^{k} \frac{1}{b_s} - \sum_{s=1}^{k} \frac{a_s}{b_s}, \tag{4}$$

where k satisfies

$$a_1 \leqslant \ldots a_k < t^* \leqslant a_{k+1} \ldots \leqslant a_n. \tag{5}$$

PROOF. The travel time t^* through used routes is the Langrangian multiplier that corresponds to the restriction (2) of optimization program (1)–(3) [9, 12, 13]. According to [9] the following relation holds:

$$t^* = \frac{F + \sum_{s=1}^{k} \frac{a_s}{b_s}}{\sum_{s=1}^{k} \frac{1}{b_s}},$$

and, hence, (4) follows directly when k satisfies (5). □

Therefore, if we know travel time of the vehicle on any of alternative routes between the OD-pair, the appropriate traffic demand can be uniquely reconstructed. Due to such results the developed approach seems to be promising. The main idea of the method based on the first principle of Wardrop: if we define the journey time of the vehicle on any of actually used routes between certain OD-pair, then we believe that the journey time on all other used routes is the same.

3. DUAL FORMULATION OF TAP

Let us consider the network of general topology presented by graph $G = (N, A)$. We introduce the following notation: W is the set of OD-pairs, $w \in W$, $W \in N$; K^w is the set of routes connecting OD-pair w; F^w is the traffic demand for OD-pair w, $F = \left(F^1, \ldots, F^{|W|}\right)^{\mathrm{T}}$; f_k^w is the traffic flow on the route $k \in K^w$, $\sum_{k \in K^w} f_k^w = F^w$; $f^w = \{f_k^w\}_{k \in K^w}$ and $f = \{f^w\}_{w \in W}$; x_a is the traffic flow on the arc $a \in A$, $x = (\ldots, x_a, \ldots)$; $t_a(x_a)$ is the link travel cost on the arc $a \in A$; $\delta_{a,k}^w$ is the indicator: 1 if the acr a is included in the route k, 0 otherwise.

User equilibrium on the transportation network G is reached by such x^* that

$$Z(x^*) = \min_x \sum_{a \in A} \int_0^{x_a} t_a(u)du, \tag{6}$$

subject to

$$\sum_{k \in K^w} f_k^w = F^w, \quad \forall w \in W, \tag{7}$$

$$f_k^w \geqslant 0, \quad \forall w \in W, \tag{8}$$

$$\tag{9}$$

with definitional constraints

$$x_a = \sum_{w \in W} \sum_{k \in K^w} f_k^w \delta_{a,k}^w, \quad \forall a \in A. \tag{10}$$

User equilibrium principle allows us to introduce t_w^*, that is equilibrium journey time for any OD-pair w.

Proposition. t_w^* is the Lagrange multiplier in the optimization program (6)–(10) corresponding to the constraint (8).

Proof. The Lagrangian of the problem (6)–(10) is

$$L = \sum_{a \in A} \int_0^{x_a} t_a(u)du + \sum_w \mu_w \left(F^w - \sum_{k \in K^w} f_k^w \right) + \\ + \sum_w \sum_{k \in K^w} \eta_k^w \left(-f_k^w\right),$$

where μ_w and $\eta_k^w \geq 0$ are Lagrangian multipliers, and differentiation of the Lagrange]ian yields:

$$\frac{\partial L}{\partial f_k^w} = \sum_{a \in k} t_a(x_a) - \mu_w - \eta_k^w = 0.$$

Note, that according to complementary slackness $\eta_k^w f_k^w = 0$. Thus, for $f_k^w > 0$ the following expression holds

$$\sum_{a \in k} t_a(x_a) = \mu_w, \quad \forall k \in K^w, w \in W \tag{11}$$

Actually, the left part of (11) is the journey time on any used route ($f_k^w > 0$) between OD-pair r. Therefore, proposition is proved.

Eventually, according to the proposition the following equality is true:

$$t_w^* = \sum_{a \in k} t_a(x_a) \quad \forall k \in K^w, w \in W.$$

We introduced multipliers $T = (t_1, \ldots, t_{|W|})^{\mathrm{T}}$ for the constraints (7), and define the dual traffic equilibrium problem:

$$\max \theta(T)$$

where $\theta(T)$ is defined by

$$\theta(T) = \min_{f \geqslant 0} \left\{ \sum_{a \in A} \int_0^{x_a} t_a(s)ds + \sum_r t_r \left(F^r - \sum_{k \in K^r} f_k^r \right) \right\},$$

subject to definitional constraints

$$x_a = \sum_{w \in W} \sum_{k \in K^w} f_k^w \delta_{a,k}^w, \quad \forall a \in A.$$

4. OD-MATRIX ESTIMATION FROM PLATE SCANNING SENSORS

Link-flow counts provide the amount of vehicles on the links. Plate scanning sensors associated with the certain links identify plates of vehicles from link-flow. Thus, when any vehicle crosses a link with some sensor then sensor records its plate number and fixation time. Eventually, database consisting of {plate number, fixation time, number of sensor} is accumulated [3]. With the help of such database, the travel time between any origin-destination pair can be directly evaluated. Indeed, one just has to know fixation time of the vehicle in origin and fixation time in destination to define t_r^* for any r.

Therefore, the following bi-level optimization program can be formulated:

$$\min_F \left(\overline{F} - F\right)^{\mathrm{T}} U^{-1} \left(\overline{F} - F\right) + (T^* - T)^{\mathrm{T}}(T^* - T), \tag{12}$$

subject to

$$F \geq 0, \tag{13}$$

where T solves

$$\max \theta(T), \tag{14}$$

where $\theta(T)$ is defined by

$$\theta(T) = \min_{f \geqslant 0} \left\{ \sum_{a \in A} \int_0^{x_a} t_a(s)ds + \sum_r t_r \left(F^r - \sum_{k \in K^r} f_k^r \right) \right\}, \tag{15}$$

subject to definitional constraints

$$x_a = \sum_{w \in W} \sum_{k \in K^w} f_k^w \delta_{a,k}^w, \quad \forall a \in A. \tag{16}$$

Here, (12) is the generalized least squares estimation and $\overline{F}$ is the aprior volume of travel demand between all OD-pairs, and U is the weighting matrix.

5. COMPUTATIONAL EXPERIMENT

Let us consider the road network of Saint-Petersburg (fig. 2). We define seven origin-destination pairs with seven shortest routes from seven periphery origins {1,2,3,4,5,6,7} to the center destination {8}. According to STSI (State Traffic

Figure 2: Selected OD-pairs on the Saint-Petersburg road network with the shortest routes

Safety Inspectorate), nowadays there are 253 plate scanning sensors observing the road network of Saint-Petersburg (fig. 3). Due to these sensors, we are able to identify travel

Figure 3: Sensors location on the Saint-Petersburg road network

time between chosen OD-pairs (table 1). The developed approach is based on the user equilibrium principle, which suggests that the value of travel time on the shortest route is

Table 1: Journey times obtained from plate scanning sensors

Route between OD-pair	Travel time t^* (minutes)
1–8	89
2–8	80
3–8	83
4–8	78
5–8	45
6–8	57
7–8	36

the travel time on any actually used route. Moreover, we are able to calculate an aprior flow $\overline{F}$ using the gravity model [4].

Let us use these data as inputs for bi-level optimization program (12)–(16). MATLAB was employed to carry out the simulation. The results of simulation are provided in the table 2. Moreover, these results are available in comparison with aprior flows. Fig. 4 gives a visualization of such a

Table 2: Comparison of model flow with aprior flow

OD-pair	Aprior flow	Model flow
1–8	5523	5910
2–8	12232	11253
3–8	6827	6295
4–8	6938	7631
5–8	5534	5080
6–8	4254	4650
7–8	3395	3202

comparison. One can see that rough aprior estimation of

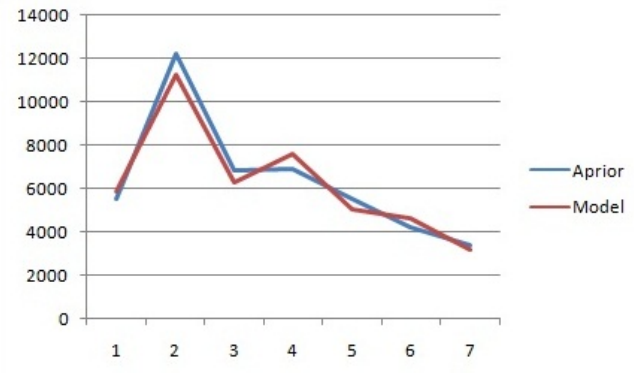


Figure 4: Comparison of model flow with aprior flow

trip flows, obtained by gravity model, was adjusted by virtue of information about actual travel time between OD-pairs. Therefore, developed in this paper approach seems to be quite useful.

6. CONCLUSION

The paper was devoted to the problem of OD-matrix estimation. The original technique of OD-matrix estimation based on a dual formulation of the traffic assignment problem was offered. Traffic demand between certain OD-pair was estimated due to the journey time obtained from plate scanning sensors. Moreover, the functional relationship between the traffic demand and the journey time was obtained explicitly for the network of parallel routes with one OD-pair. Eventually, the developed method has been tested on the experimental data of the Saint-Petersburg road network.

7. REFERENCES

[1] Hazelton M. Inference for origin-destination matrices: estimation, prediction and reconstruction // Transportation Research Part B. 2001. No 35. P. 667–676.

[2] Yang H., Sasaki T., Iida Y., Asakura Y. Estimation of origin-destination matrices from link traffic counts on congested networks // Transportation Research Part B. 1992. No 26 (6). P. 417–434.

[3] Zakharov V., Krylatov A. OD-matrix estimation based on plate scanning // 2014 International Conference on Computer Technologies in Physical and Engineering Applications (ICCTPEA). 2014. P. 209-210.

[4] Medina A., Taft N., Salamatian K., Bhattacharyya S., Diot C. Traffic matrix estimation: existing techniques and new directions // Computer Communication Review – Proceedings of the 2002 SIGCOMM conference. 2002. No 32. P. 161–174.

[5] Castillo E., Menedez J. M., Jimenez P. Trip matrix and path flow reconstruction and estimation based on plate scanning and link observations // Transportation Research Part B. 2008. No 42. P. 455–481.

[6] Bianco L., Cerrone C., Cerulli R., Gentili M. Locating sensors to observe network arc flows: exact and heuristic approaches // Computers and Operation Research. 2014. No 46. P. 12–22.

[7] Minguez R., Sanchez-Cambronero S., Castillo E., Jimenez P. Optimal traffic plate scanning location for OD trip matrix and route estimation in road networks // Transportation Research Part B. 2010. No 44. P. 282–298.

[8] Bierlaire M. The total demand scale: a new measure of quality for static and dynamic originŰdestination trip tables // Transportation Research Part B. 2002. No 36. P. 282–298.

[9] Krylatov A. Yu. Optimal strategies for traffic flow management on the transportation network of parallel links // Vestnik of St. Petersburg State University. Series 10. Applied Mathematics. Computer Science. Control Processes. 2014. No 2. P. 121–130.

[10] Patriksson M. The Traffic Assignment Problem: Models and Methods. Utrecht, Netherlands: VSP, 1994. 223 p.

[11] Wardrop J. G. Some theoretical aspects of road traffic research // Proc. Inst. Civil Eng. 1952. Vol. 1, No 3. P. 325–362.

[12] Zakharov V. V., Krylatov A. Yu. Transit Network Design for Green Vehicles Routing // Advances in Intelligent Systems and Computing. 2015. Vol. 360. P. 449–458.

[13] Zakharov V. V., Krylatov A. Yu. Competitive routing of traffic flows by navigation providers // Automation and Remote Control, 2016. Vol. 77, No 1. P. 179–189.

WWW-based System for LEGO Robots Control

Georges Meguro
University of Aizu
Tsuruga, Ikki-machi
Aizu-Wakamatsu, Japan
+81-242-372717
m5201110@u-aizu.ac.jp

Alexander Vazhenin
University of Aizu
Tsuruga, Ikki-machi
Aizu-Wakamatsu, Japan
+81-242-372717
vazhenin@ u-aizu.ac.jp

ABSTRACT
This article proposes a WEB-based LEGO Robot control system with JAVA technologies. This work is based on the original Virtual-Model-View-Controller design pattern, and focused to exchange data between server and robot. The designed system can manage the robot and reflects the collected data by robot to the client on web page.

Categories and Subject Descriptors
I.2.9 [**Artificial Intelligence**]: Robotics – *operator interfaces, sensors, workcell organization and planning*.

General Terms
Algorithms, Design, Experimentation

Keywords
LEGO Robots, WWW interface, Data Exchange Protocol.

1. INTRODUCTION
The current state of research in robotics has moved from a limited capability single robot to multi-robot systems equipped with a plethora of sensors, leading to de-facto multi-processor and distributed applications. Developing ad-hoc solutions based on low level communication libraries is not an efficient approach and makes reuse and sharing of software difficult. Software frameworks can be considered as an important paradigm for the next generation of robotics applications [1]. Actually, robots need to use information from many sources. For example, the scan system recognizes objects, a mapping system builds an environment map, a human may give commands, and the robot loads data and knowledge from different sources. For making use of this knowledge, a robot needs to integrate these different pieces of information, represent them in a common format and language including its semantics. Recent robotics studies are investigating how robots can exploit the World Wide Web in order to offer their functionality and retrieve information that is useful for completing their tasks [2, 3]. The effective way to realize this is represented by the Service Oriented Architecture (SOA), where the Service Component Architecture defines a component-based implementation [4]. Robotic development environments leverage these emerging Web-programming paradigms [5].

Our investigations are based on the original Virtual-Model-View-Controller (V-MVC) design pattern that is a combination of two well-known approaches: SOA and the Model-View-Controller (MVC). Currently, we are designing the VMVC-based technology for developing SOA-applications that reduces the developer's efforts by concentrating mostly on creating the processing logic and the UI design, facilitating the debugging process of the view and model separately [6]. The goal of this research is to develop WEB-based applications for a Robot Control System, visualization of robot activities and data in the framework of Service-Oriented Architecture and the VMVC-based technology.

LEGO Education EV3 robotics is the third generation of LEGO Educational robotics and comes with an improved intelligent brick, new motors, new sensors and improved software. The best solution is to teach students how to program, and then to build and test their models. The first visual programming environment was called LEGOsheets, and was created by the University of Colorado in 1994.

The work presented is devoted to designing the WEB-based Robot control system in the framework of the Apache Tomcat environment. This system is an open source software implementation of the Java Servlet, Java Server Pages, Java Expression Language and Java WebSocket technologies. During this work a Robot control server was designed allowing communication with an internal robot program, accumulating data and visualizing them. More specifically, the paper is devoted to designing the mechanism of data exchange between a WEB server and LEGO robot controlled by the client side.

This paper has the following structure. Section 2 contains the description of the environment, which is used for creating the system. Explanation about the data exchange mechanism design is presented in Section 3. Results of testing are presented in Section 4. The conclusion and future work are discussed in Section 5.

ICAIT'16, Oct. 6–8, 2016, Aizu-Wakamatsu, Japan.

2. LEGO ENVIRONMENT

2.1 Hardware

LEGO MINDSTORMS is a robot construction kit which is designed by Lego. It's constructed mainly from a computer brick, which can run programs, various sensors for gathering data, and a motor for dynamic force (Fig. 1).

EV3 is the third generation product of LEGO MINDSTORMS. Its computer brick has an ARM9-type CPU, 64 MB of RAM memory and 16 MB of flash memory, MicroSDHC card slot; Linux is installed. In addition, EV3 also supports Bluetooth and Wi-Fi for wireless communication [7].

Figure 1. Constructed LEGO MINDSTORMS EV3

2.2 LeJOS Firmware

LeJOS is a firmware replacement for LEGO MINDSTORMS. It includes a Java virtual machine (Java Runtime Environment 7) and allows Java program execution on LEGO MINDSTORMS. This means it can work with other Java-based technology. Typically it includes:

- A library of Java classes (classes.jar) that implement the LeJOS EV3 Application Programming Interface (API) and provides an alternative Java Runtime (packages java.*) that is optimized for the EV3.
- A linker for linking user Java classes with classes.jar to form a binary file that can be uploaded and run on the EV3.
- PC tools for flashing the firmware, uploading programs, debugging, and many other functions.
- A PC API for writing PC programs that communicates with LeJOS EV3 programs using Java streams over Bluetooth or USB, or using the LEGO Communications Protocol (LCP).

In this research we are using the LeJOS EV3 0.9.0-beta version [8].

3. APPLICATION SCENARION

3.1 Architecture

Usually, a WEB-application is a client-server software application in which the client (or user interface) runs in a web browser. There are several ways to include a robot into a client-server environment. In our investigations we are using the thin client approach in which a robot has direct connection with the server. This architecture is shown in Figure 2.

In order to establish this web-architecture, we are using the Apache Tomcat system that is an open source software implementation of the Java Servlet, Java Server Pages, Java Expression Language and Java WebSocket technologies. We are using these technologies to create a web application in conjunction with the Lejos Java program. In the presented work the Apache-Tomcat-8.0.28 is used [9].

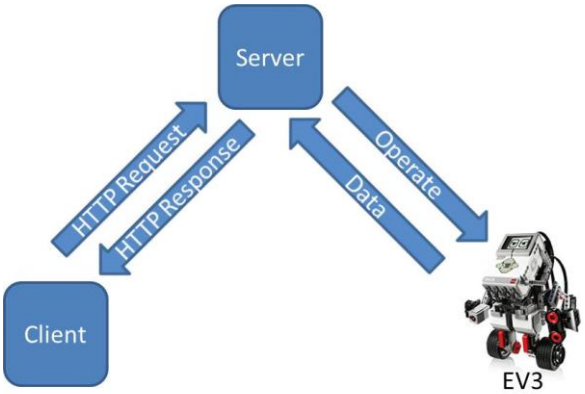


Figure 2. Basic WEB-architecture

3.2 Application Scenario

Consider the following application of the LEGO robot. It is oriented to scan designated areas, accumulate some data from the LEGO sensors, collect these data at the server, and show results on the client site as well as control robot movement and actions. An example of area is shown in Figure 3.

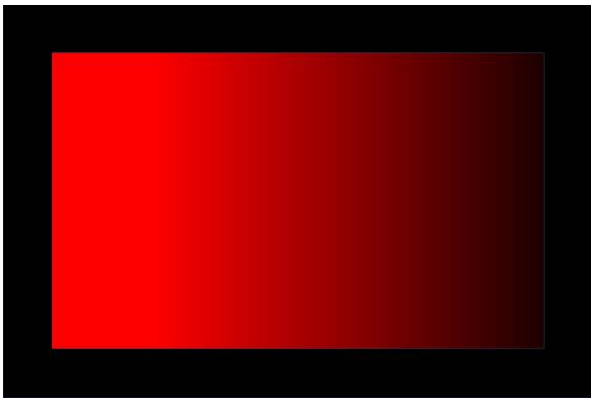


Figure 3. Scanning Area

Importantly, a robot can have different algorithms for its movement defined by the EV3 program. For example, the LEGO robot can move across this area with a zigzag trajectory by changing its direction after reaching area boundaries.

Figure 4 shows a sequence diagram realizing the implementation scenario as follows. First, the application is launched by client request to an application server. Next, the server connects to EV3 and orders it to start exploration. Then, EV3 sends the explored data to the server. Finally, the server generates the web page to show the result and sends it to the client.

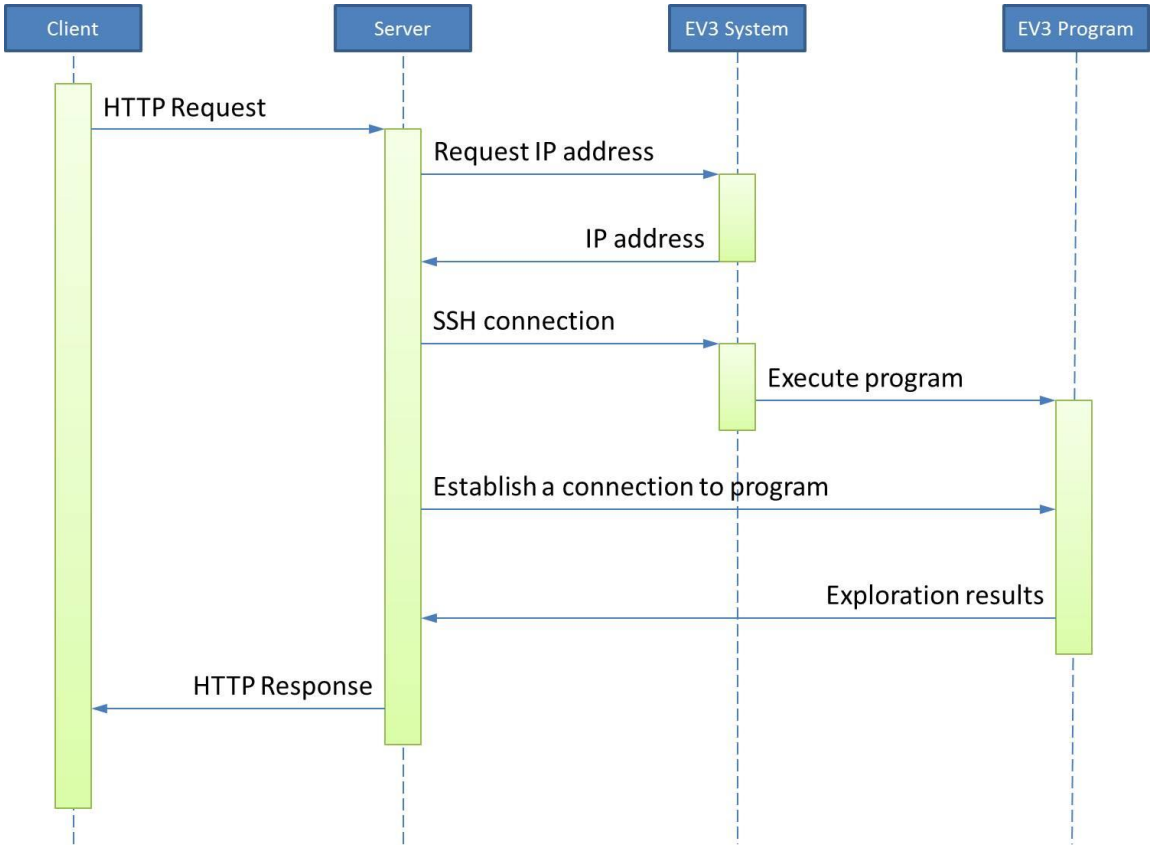


Figure 4. Implementation Diagram

3.3 Communication between Server and LEGO Robot

The Wi-Fi network allows high-speed connections between devices supporting on-line control for the robot behavior and data exchange. The efficient way to support security of Wi-Fi connections is in using encryption of exchanged data. SSH-protocol provides a secure channel over an unsecured network in client-server architecture. Accordingly, the communication protocol between server and robot is realized as follows. First, the program in the server searches connectable EV3 and obtains its IP address. Next, the server connects to EV3 by SSH using obtained an IP address to launch program in EV3. Finally, each program makes a connection through Java socket and starts exchanging data to solve task.

To realize this protocol, it is necessary to grant remote access from one device to operate on other device from the server, run programs or manipulate with files. We realized it using the Ganymed SSH-2 for Java that is a library which implements the SSH-2 protocol in pure Java and realizes connecting to other devices by SSH from Java programs. It supports SSH sessions (remote command execution and shell access), local and remote port forwarding, local stream forwarding, X11 forwarding, SCP and SFTP. There are no dependencies on any JCE provider, as all crypto functionality is included. Ganymed-ssh2-build210 is used in this paper [10]. Figure 5 describes the basic Java code implementing a connection between a server and an external device.

```
//Connect to device and log in.
Connection conn = new Connection(IPADDRESS);
boolean connected
      = conn.authenticateWithPassword(USERID, PASSWORD);

//Execute the command line at logged-in device
if (connected) {
   Session session = conn.openSession();
   session.execCommand(COMMANDLINE);
   session.close();
}
conn.close();   //Disconnect
```

Figure 5. Connecting to Device via the Ganymed SSH-2 Library

4. IMPLEMENTATION AND TESTING

To test the communication protocol designed, a WWW-based prototype was developed. It has a rather simple end-user interface allowing operation to be launched inside the robot from a browser. The current version of the program allows having access to the color robot sensors.

Figure 6 shows a flowchart of a prototype EV3 program which operates the robot to explore the field shown in Figure 3, and

send the red density of the current position to the server. As shown in Figure 3, the area has a red colored area with different color density. It also bordered by black colored lines.

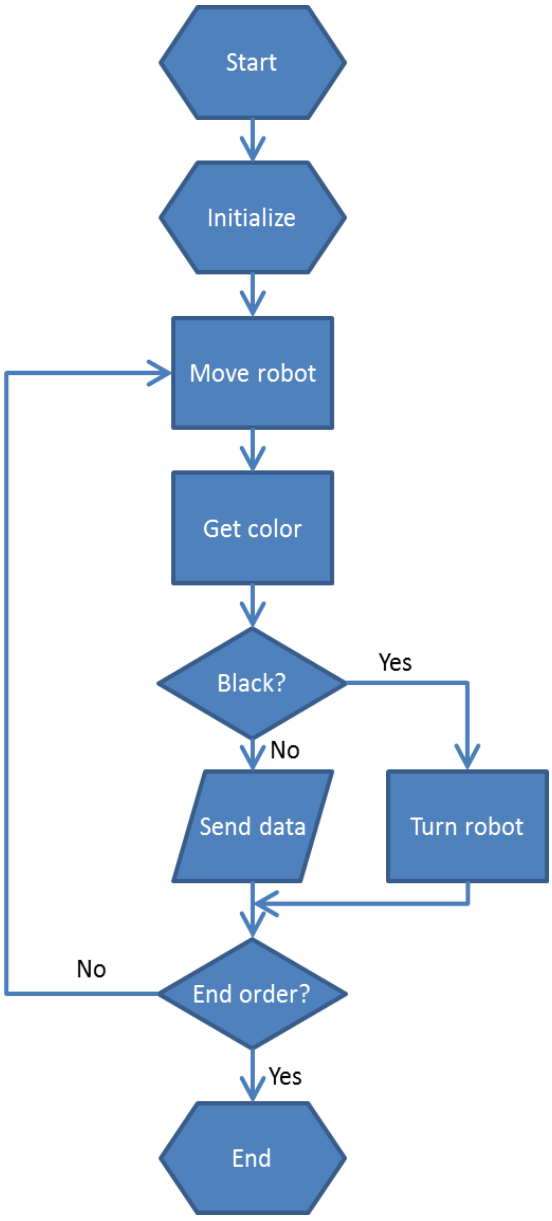


Figure 6. Flowchart of prototype program

The program execution is started from the initialization step. During initialization, the program makes a connection to the server using a Java socket. It also collects information about the robot motor and sensor, which is connected to the computer brick and will participate in operation. After the initialization, the robot starts movement in a straight direction. At the same time, it starts the acquisition of data from the color sensor. Acquired color information is transmitted to the server.

If the sensor detects the black line of the outer edge, the robot starts rotation to avoid jumping out from the field. The program keeps exploring until the server sends the stop instruction.

By moving the LEGO robot over the designated area, it is possible to get codes of corresponding colors and show them in the browser's window. The experiments provided confirm the correctness and accuracy of the designed protocol.

5. CONCLUSION

The work presented is devoted to designing a WEB-based Robot control system in the framework of the Apache Tomcat environment. More specifically, the mechanism of data exchange between a WEB server and LEGO robot was designed and tested. The experiments provided confirm the correctness and accuracy of the designed protocol. The communications between server and the EV3 program were realized via SSH-protocol that provides a secure channel over an unsecured network in the client-server architecture. The experiments provided confirm the correctness and accuracy of the designed protocol.

This work can be considered as a first step in designing the SOA-based Robot control system based on a V-MVC paradigm. In future work, we are planning to improve the test program and create a V-MVC model of this system.

6. REFERENCES

[1] Chugo D. and Yokota S. (Eds). 2012. *Introduction to Modern Robotics II. Syst.* iConcept Press Ltd.

[2] Blake M.B., Remy S.L., Wei Y. and Howard A. M. 2011. Robots on the Web. *IEEE Robotics & Automation Magazine* (June 2011), 58-68.

[3] M. Tenorth and Ul. Klank, D. 2011. Pangercic, and M. Beetz, Web-Enabled Robots. IEEE Robotics & Automation Magazine (June 2011), 33-43.

[4] Erl Th. 2009. *SOA Design Patterns*. PrenticeHall.

[5] Remy S.L. and Blake M.B. 2011. Distributed service-oriented robotics. *IEEE Internet Comp.* 5, 2 (Mar./Apr. 2011), 70–74.

[6] Cortez R. and Vazhenin A. 2015. Virtual model-view-controller design pattern: Extended MVC for service-oriented architecture. *IEEJ Trans. on Electrical and Electronic Engineering* 10 (July 2015), 411-422.

[7] Lego Mindstorms - LEGO.com. http://mindstorms.lego.com/.

[8] Lejos, Java for Lego Mindstorms. http://www.lejos.org/.

[9] Apache Tomcat. http://tomcat.apache.org.

[10] The Ganymed Project. http://www.ganymed.ethz.ch/ssh2.

Visual Learning Tool for Lego Robots Programming

Georges Meguro
University of Aizu
Tsuruga, Ikki-machi
Aizu-Wakamatsu, Japan
+81-242-372717
m5201110@u-aizu.ac.jp

Kenji Abe
University of Aizu
Tsuruga, Ikki-machi
Aizu-Wakamatsu, Japan
+81-242-372717
s1200222@u-aizu.ac.jp

Alexander Vazhenin
University of Aizu
Tsuruga, Ikki-machi
Aizu-Wakamatsu, Japan
+81-242-372717
vazhenin@u-aizu.ac.jp

ABSTRACT
This article proposes visual learning tool for Lego MINDSTORMS EV3 using LEJOS firmware. It has an icon-based style of introducing robot activities with on-line reflection of corresponding code. This allows beginners to enjoy and make EV3 programs. The designed tool can launch and control the robot actions and check/debug them even if a program isn't complete. It also enabled that a client operates on the web page and moves a robot through a server.

Categories and Subject Descriptors
D.1.7 [**Visual Programming**]

General Terms
Design, Human Factors, Languages.

Keywords
Visual Programming, Robotics, Icon-based Language.

1. INTRODUCTION
The current state of research in robotics has moved from a limited capability single robot to multi-robot systems equipped with a plethora of sensors, leading to de-facto multi-processor and distributed applications. Developing ad-hoc solutions based on low level communication libraries is not an efficient approach and makes reuse and sharing of software difficult. Software frameworks can be considered as an important paradigm for next generation of robotics applications [1]. Actually, robots need to use information from many sources. For example, the scan system recognizes objects, a mapping system builds an environment map, a human may give commands, and the robot loads data and knowledge from different sources. For making use of this knowledge, a robot needs to integrate these different pieces of information, represent them in a common format and language including its semantics. Recent robotics studies are investigating how robots can exploit the World Wide Web in order to offer their functionality and retrieve information that is useful for completing their tasks [2, 3]. The effective way to realize this is represented by the Service Oriented Architecture (SOA), where the Service Component Architecture defines a component-based implementation [4]. Robotic development environments leverage these emerging Web-programming paradigms [5].

Our investigations are based on the original Virtual-Model-View-Controller (V-MVC) design pattern that is a combination of two well-known approaches: SOA and the Model-View-Controller (MVC). Currently, we are designing the VMVC-based technology for developing SOA-applications that reduces the developer's efforts concentrating mostly on creating the processing logic and the UI design, facilitating the debugging process of the view and model separately [6]. The goal of this research is in developing WEB-based applications for Robot Control System, visualization of robot activities and data in the framework of Service-Oriented Architecture and the VMVC-based technology.

LEGO Education EV3 robotics is the third generation of LEGO Educational robotics and comes with an improved intelligent brick, new motors, new sensors and improved software. It can be considered as a perfect solution allowing to teach students how to program, build and test their robot models. The first visual programming environment was called LEGO sheets [1]. It was created by the University of Colorado in 1994.

The paper presented is devoted to design a visual tool allowing users study how to program robot activities. It has an icon-based style of introducing robot activities with on-line reflection of corresponding code. This allows beginners to enjoy and make the EV3 program. The designed tool can launch and control the robot actions and check/debug them even a program isn't complete. It also enabled that a client operates on the web page and moves a robot through a server.

The organization of this paper is as follows. Section 2 describes the Lego MINDSTORMS environment and proposes the way of improving it. Section 3 describes the proposed application. Section 4 describes the visual robot program made with the application and the experimental results of its testing. Section 5 contains conclusion and future directions of work.

ICAIT'16, Oct. 6–8, 2016, Aizu-Wakamatsu, Japan.

2. LEGO ROBOT ENVIRONMENT

2.1 Lego MINDSTORMS

Lego MINDSTORMS is an educational toolkit for robotics that is assembled with Lego Blocks. It is oriented to study how to construct and program different kinds of LEGO robots with some enjoyment (Figure 1). Now, there is the third generation of this toolkit called RCX, NXT and EV3 correspondingly. EV3 is the latest model of Lego MINDSTORMS. It contains four sensor ports, four motor ports, an USB port and Micro SD slot. It may also be extended by a Wi-Fi function allowing to make the wide range of communications in comparison with the NXT version. Table 1 shows some characteristics of the Lego MINDSTORMS EV3.

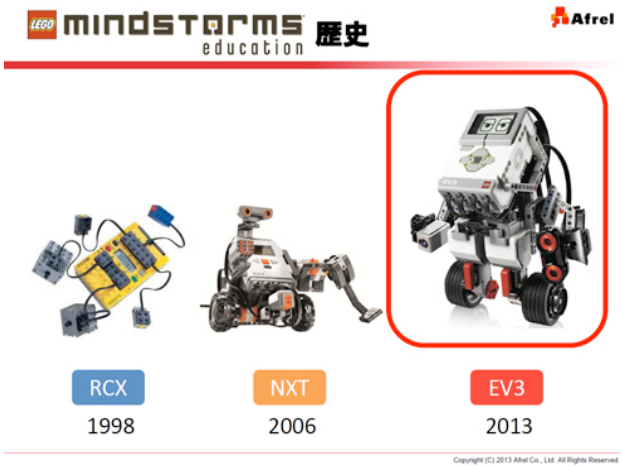


Figure 1. LEGO MINDSTORMS

Table 1. Spec of Lego MINDSTORMS EV3

OS	LINUX
Main Processor	300MHz
Main Memory	64MB
Storage	16MB

2.2 Firmware

The educational software called EV3 is based on LabVIEW of National Instruments Software [7, 8]. The LEGO programming is realized by connecting icons using drag and drop manipulations. Additionally, each icon can have numerical data defining physical parameters of robots like speed, angle, time, etc.

Besides this software, EV3 can also use several programming platforms based on C and/or JAVA languages. The LEJOS is the extension of JAVA language for the Lego MINDSTORMS programmable bricks [9]. To run the LEJOS firmware, it is necessary to prepare the micro SDHC memory card with at least 2GB, and no more than 32GB. The card needs to be formatted with a FAT32 partition.

2.3 Problem

EV3 Software allows making a LEGO program visually and easily without typing the text. So, it isn't necessary to type, and even beginners can be interested. But it doesn't have versatility especially in case when it is necessary to study API and program constructions at the same time.

Conversely, LEJOS uses JAVA language showing a program code directly and allowing to learn basic language constructions in textual mode. Therefore, it makes some difficulties for beginners who is going to study the low level of the LEGO programming. That is why it would be good to have tools combining advantages of visual programming with possibility to study how to specify LEGO robot operations by means of Java API interface.

3. ICON-BASED VISUAL PROGRAMMING TOOL

The presented tool was designed to support the LEGO programming and study combining the icon-based specification of LEGO robot operations with simultaneous reflection of the Java executable code.

The programming panel is shown in Figure 2. Actually, the programming process is reduced to choosing a corresponding icon and specifying its parameters. It can also display a code of EV3 program in real time. This allows learning how to make a sequence of Java instructions with a corresponding reflection of the operation semantics. It is also possible to run the robot program and watch immediately how it works.

After finishing, editing and debugging, the user can save a robot program as a Java code. In future, this code can be edited as an ordinary Java program.

System is saving both the Java code and a list of visual blocks in order to have a correspondence between visual and textual part of a robot algorithm. There are the following panels in the system interface:

1 ***Message Panel*** displays errors and message of preservation completion.

2 ***Port Panel*** allows choosing motor ports and sensor ports.

3 ***Program Table Panel*** displays a code of visual program. After pressing the "RUN" button, program sequence will be sent to the EV3 robot.

4 ***Program Sentence Panel*** displays a Java program sentence. The user can save a program as a JAVA file.

5 ***Program Button Panel (If and Loop)*** allows selecting and adding/removing program blocks. It supports has if- and loop- statement blocks.

6 ***Program Button Panel*** allows choosing/adding program blocks including the blocks' movement.

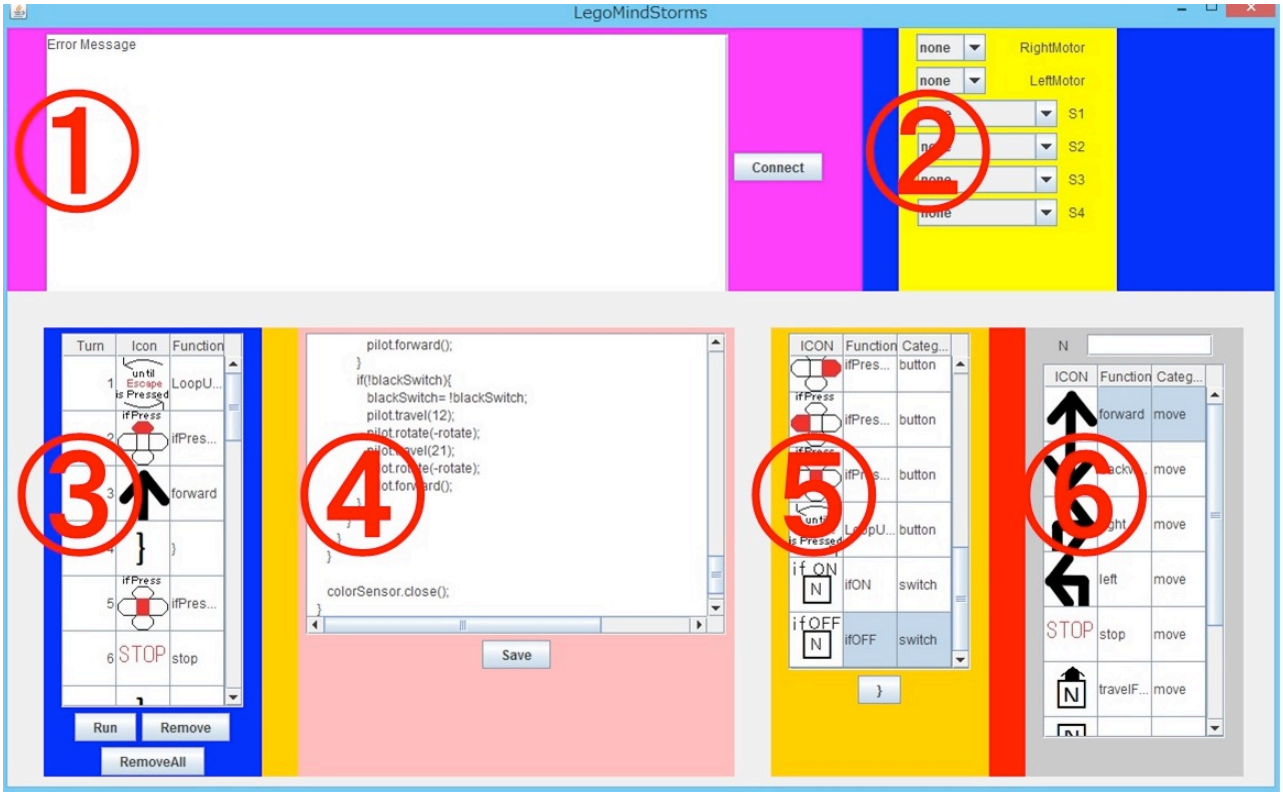


Figure 2. Programming panels of the EV3 visual editor

Figure 3 shows the Data Flow Process that demonstrates how data are transformed in the environment presented.

This application saves data in an array list including the visual blocks list named a program table and the Java code named a program sentence. The user can change the program table by inserting/removing a visual block and specifying its parameters. Synchronously, the corresponding changes will be made in the program sentence. At any time, the user can launch execution of the robot program. In this case, the program sentence is loading into EV3, and the editor will launch its execution.

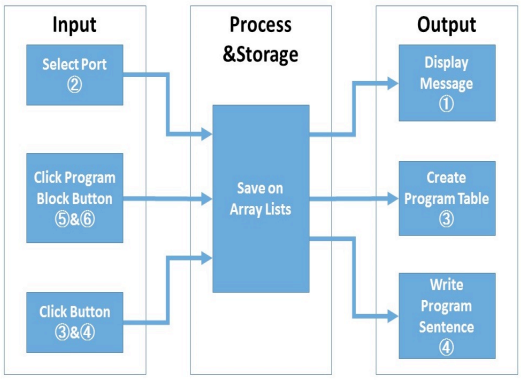


Figure 3. Input-output data flow

The robot program can be executed via a stand-alone and WEB-based application correspondingly. In both cases the connection between EV3 and a host is realized using WI-FI connections.

4. EXAMPLE OF APPLICATION

The features of the presented system were tested using the algorithm that is oriented to find a part with the given color on the rectangular area as shown in Figure 4. The LEGO robot is moving across this area by a zigzag trajectory. The area is bounded by a black color, and a robot is changing its direction as shown in Figure 4. During this movement, the robot saves and displays colors of all areas visited. The movement will be stopped when the area with given color is reached.

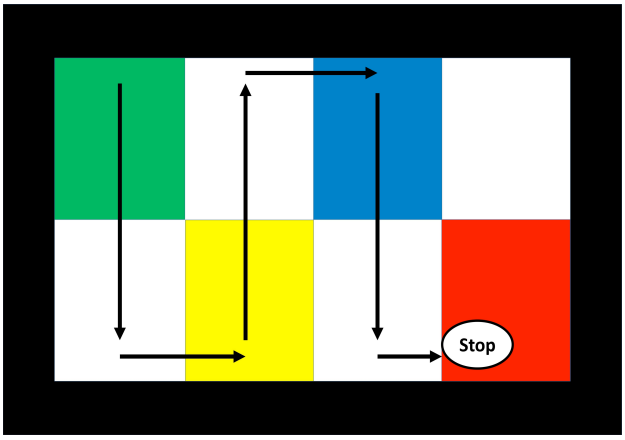


Figure 4. The route where EV3 runs

The block diagram of this algorithm is presented in Figure 5. As shown in Figure 6, the visual program realizing this robot's behavior consists of twenty-four visual blocks. It shows practical examples of usage the icon language especially IF-statement allowing programming according to states of the robot sensors. Figure 7 illustrates the program logic and shows how visual constructions are nested. Obviously, in our experiments we used the color LEGO sensor allowing to distinguish sixteen different colors.

Figures demonstrate examples with the red given color and black colored boundaries. At first, EV3 moves forward. Robot stops its movement when the area with given color is reached. It changes direction when a black boundary will be found and continues movement. It implements counterclockwise or clockwise turns realizing the zigzag movement.

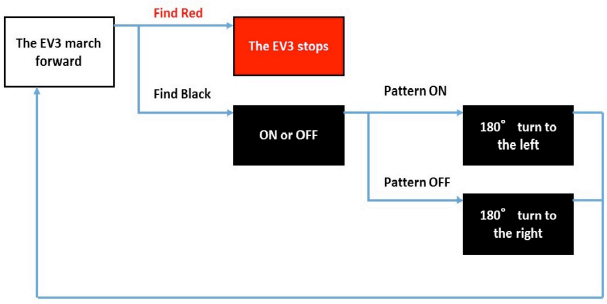


Figure 5. Find color program algorithm

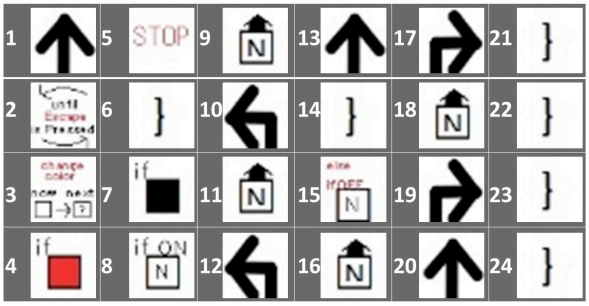


Figure 6. Icon of Find color program algorithm

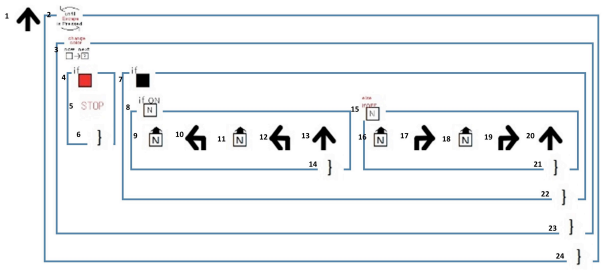


Figure 7. Visual blocks of Find Color Program

The size of the area is an important parameter influencing on the search results. This requires changing the width of zigzag. Figure 6 demonstrates also possibility to specify time of forward movement. Other experiments were provided to evaluate possibility to find area with sizes of 18, 6 and 3 centimeters. On-line changing of this parameter allowed us to optimize this search procedure.

5. CONCLUSION

In this research, we propose a new application based on a Java language which enables to program visually. It allows to make a program by selecting icons and specifying physical parameters of each operation. The system generates a Java-program that can be loaded in the LEGO robot. The main feature of the system is that any visual object is immediately displayed by representing a corresponding Java code. It is also possible to confirm so how the sentence is written. The presented work is made on cooperation with another research oriented to make robot control program via WEB browser.

To test the system, we asked six participants. It was confirmed the simplicity of programming and on-line study process. This opinion confirms workability the program presented and defines future directions in its evolution based on some improvement of the user interface.

6. REFERENCES

[1] D. Chugo and S. Yokota (Eds). 2012. *Introduction to Modern Robotics II. Syst.* iConcept Press Ltd. 294.

[2] M. B. Blake, S. L. Remy, Yi Wei, and A. M. 2011. Howard, Robots on the Web. *IEEE Robotics & Automation Magazine* (June 2011). 58-68.

[3] M. Tenorth and Ul. Klank, D. 2011. Pangercic, and M. Beetz, Web-Enabled Robots. *IEEE Robotics & Automation Magazine* (June 2011). 33-43.

[4] Th. Erl. 2009. *SOA Design Patterns*. PrenticeHall.

[5] S. L. Remy and M. B. Blake. 2011. Distributed service-oriented robotics. *IEEE Internet Comput* 15/2, (Mar./Apr. 2011). 70–74.

[6] R. Cortez and A. Vazhenin. 2015. Virtual model-view-controller design pattern: Extended MVC for service-oriented architecture. *IEEJ Trans. On Electrical and Electronic Engineering* 10 (July 2015), 411-422.

[7] *Ltd Afrel, Educational Lego Mindstorms regular agency*. Ltd. Afrel. http://www.afrel.co.jp/lineup/mindstorm-ev3/.

[8] 2014. *Education version EV3 Java programming guide.* Ltd. Afrel.

[9] *LEJOS, Java for Lego Mindstorms.* Lejos. http://www.lejos.org/.

[10] 2013. *LEGO MINDSTORMS EV3*. LEGO group. 4-31.

[11] R. Cortez, and A. Vazhenin. 2013. Developing Re-usable Components Based on the Virtual-MVC Design Pattern. *Databases in Networked Information Systems*. 132-149.

[12] N. Teslya, and S. Sergey. Smart-M3-based robot interaction in cyber-physical systems. *Open Innovation* (FRUCT16, 2014 16th Conference of. IEEE). 108-114.

Assessing Similarities in Soccer Team Tactics

Akitaka Moriyama Maxim Mozgovoy
The University of Aizu
Tsuruga, Ikki-machi,
Aizuwakamatsu, Fukushima, Japan
+81-242-37-2664
m5201103, mozgovoy@u-aizu.ac.jp

ABSTRACT

We describe similarity assessment of soccer game formations between teams by using two different approaches: (a) a sum of distances between each pair of assigned players obtained with Hungarian algorithm; (b) a cosine similarity between team heat maps. We test our algorithms separately for offense and defense strategies of teams using real soccer game data. As a result, by considering similarities we can reveal similar and distinct approaches to attack and defense used by different soccer teams.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Theory

Keywords

team similarity, classification, clustering

1. INTRODUCTION

The importance of data analysis on sports had been increasing year by year. Soccer game is not an exception. The outcome of a game depends on players and their behavior on the field. Skills and strategies are advanced as time goes on, so detailed analysis of team tactics becomes and important problem in sport analytics [1].

However, prediction of group games of player action and trends of the team such as soccer game is not easy. One of the problems is the difficulty of quantitative analysis of player abilities. Thus, research of soccer game data is scarce.

In this paper, we deal with formation analysis by comparing teams using real game data of soccer games. The data is taken from actual J1 League games (J1 League is the top division of Japan professional soccer). We attempt to analyze similarities and differences between team tactics, and evaluate consistency of tactics of the same teams across games.

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

2. METHODOLOGY

We use the data of five matches, provided by Data Stadium Inc [2]. These matches were played in J1 league in 2011 [3]. There are six teams in the dataset: Koube, Nagoya, Shimizu, Yamagata, Urawa, and Yokohama. Urawa and Yamagata teams played only one game, while other teams played two games.

Each game data file contains three chunks. The first chunk contains only the current frame number. The second chunk lists the status of each player (home team or away team, jersey number, coordinates, and speed). The last chunk contains the ball status, represented by ball coordinates, speed, owner, and information whether the game is currently suspended.

At the preprocessing stage we extract only game data that corresponds to non-suspended periods of the game. We assume that team formations in game pauses is not important. Next, team formation is different between defense phase and offense phase. We divide the data into defense phase and offense phase. In addition, we divide the data into the first half, and the second half. We compare team tactics across individual halves of the game, but never mix defense and offense phases. We use two different methods of comparison:

1. Sum of distances between each assigned player obtained with Hungarian algorithm.
2. Cosine similarity between team heat maps.

Hungarian method is one of the methods for solving the assignment problem [4]. We can optimally juxtapose the players of a team in two different situations to minimize a distance between them. In the current work, we consider assignment optimal, if the sum of distances between the assigned players is minimal. We calculate each of player 's average coordinates from team data, and apply Hungarian method to two teams, we then derive the distance. We repeat it for every team. We use Euclidean distance to measure a distance between the players.

Our second method needs a heat map of each team, obtained from coordinate data of each player. We divide a soccer field into a grid of 20 by 13 cells, and calculate a probability for each cell to be occupied by some team player from a team data, and make a heat map. We then connect

map rows sequentially from top to bottom to obtain a vector. Thus a map converted into a vector. We repeat it for every team. These heat maps represented with vectors are compared using cosine similarity (a dot product between the vectors).

We calculate an average distance and similarity between each possible pair of teams separately for defense phase and offense phase.

3. EXPERIMENTAL RESULTS

Altogether, we compare 190 pairs of teams in each (offense / defense) phase. The average results are shown in the tables 1-4.

Table 1 shows the distances between team pairs obtained with Hungarian algorithm in the defense phase. It is clearly seen that the teams Koube, Urawa, and Yamagata are very consistent in their defense tactics, and show considerably smaller distances between their own games. However, Shimizu and Yokohama teams are sometimes closer to other teams in their defense style rather than to their own behavior in other games.

Table 1: Hungarian distances (defense)

	Koube	Nagoya	Shimizu	Urawa	Yamagata	Yokohama
Koube	5523.85					
Nagoya	8330.71	7905.68				
Shimizu	6322.05	7643.58	6997.47			
Urawa	7962.39	6338.40	6666.29	4436.14		
Yamagata	7509.70	6872.96	6172.73	5145.65	3953.71	
Yokohama	6909.35	6947.56	6516.39	5988.50	6091.26	6622.84

Table 2 shows the distances between team pairs obtained with Hungarian algorithm in the offense phase. In this experiment it is evident that the teams are more experimental in attack, and show higher variety of styles in different games. There is no consistency across different games of the same team (except Urawa team).

Table 2: Hungarian distances (offense)

	Koube	Nagoya	Shimizu	Urawa	Yamagata	Yokohama
Koube	5563.33					
Nagoya	7010.16	6274.00				
Shimizu	5914.80	8467.29	6466.92			
Urawa	5072.38	5318.69	6624.46	3971.05		
Yamagata	5695.01	6466.35	6171.26	4873.85	5918.03	
Yokohama	6291.78	7056.96	7189.49	6068.62	6471.06	7775.19

Table 3 shows the cosine similarities between team heat maps in the defense phase. Interestingly, heat maps of all teams are very similar, and the similarity score never drops below 0.88. Probably, a heat map better characterize the game of soccer itself rather than individual team tactics. However, this method also lists Koube and Urawa among the most consistent teams in the dataset.

Table 4 shows the cosine similarities between team heat maps in the offense phase. These values are again consistent with the values shown in Table 1: the highest similarities are exhibited by Koube, Urawa, and Yamagata teams.

4. DISCUSSION AND CONCLUSION

Table 3: Cosine similarities (defense)

	Koube	Nagoya	Shimizu	Urawa	Yamagata	Yokohama
Koube	0.95					
Nagoya	0.89	0.92				
Shimizu	0.92	0.91	0.94			
Urawa	0.92	0.92	0.93	0.95		
Yamagata	0.92	0.92	0.93	0.93	0.91	
Yokohama	0.88	0.91	0.91	0.92	0.91	0.90

Table 4: Cosine similarities (offense)

	Koube	Nagoya	Shimizu	Urawa	Yamagata	Yokohama
Koube	0.94					
Nagoya	0.88	0.90				
Shimizu	0.91	0.90	0.90			
Urawa	0.92	0.89	0.91	0.93		
Yamagata	0.90	0.91	0.91	0.91	0.94	
Yokohama	0.90	0.90	0.91	0.92	0.93	0.93

It is interesting to note that the features of individual team is more clearly visible in the defense phase of a game. Most teams adhere to similar defense strategies across different games. Offensive tactics vary highly, and teams experiment when playing against different opponents. This trend is seen both in Hungarian algorithm-based, and cosine similarity-based calculations.

In its turn, cosine similarity of heat maps cannot be considered a reliable indicator of team behavior similarity. All teams in our dataset produce very similar heat maps, probably, reflecting the nature of soccer rather than individual team tactical schemes.

The present work shows that most teams exhibit numerically detectable features that make them different from other teams. We hope that this method will contribute to data analysis and further development of soccer game research.

5. REFERENCES

[1] 加藤 健太. サッカーにおけるデータ分析とチーム強化 (data analysis and team strengthening on soccer game). **電子情報通信学会　通信ソサイエティマガジン** *(IEICE)*, 10(1):29–34, 2016.

[2] データスタジアム株式会社 (Data Stadium Inc.). `https://www.datastadium.co.jp`. Accessed: 2016-08-05.

[3] J リーグ (J League). `http://www.jleague.jp`. Accessed: 2016-08-05.

[4] James Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1):32–38, 1957.

Hadoop Environment for the Analysis of Large Network Packets

Keisuke Kato
The University of Aizu
Software Engineering Laboratory
Ikki-machi, Aizu-Wakamatsu, Fukushima
965-8580
m5191107@u-aizu.ac.jp

Vitaly Klyuev
The University of Aizu
Software Engineering Laboratory
Ikki-machi, Aizu-Wakamatsu, Fukushima
965-8580
vkluev@u-aizu.ac.jp

ABSTRACT

Nowadays, companies, governments, etc need to defend private information and data such as company's and government institution's data or latest research achievements. However, the crackers are maliciously attacking to steal the important data from them and sell the information such password, credit card number, etc. As the results of these attacks, many companies are getting seriously damages and they are loosing the trust from the users, customers, citizens, etc. Therefore, researches in computer security are required to develop intelligent defense systems. In network security, analyzing huge size of network packet capture (pcap) files is very important to monitor the behavior of networks and/or develop an intrusion detection system (IDS), intrusion prevention system (IPS), web application firewall (WAF), etc. In this paper, we developed an environment by using Hadoop. We executed SQLs to analyze about 85GB of network packet dataset provided by the University of New Brunswick. We presented how to analyze the huge size of pcap files on Hadoop and visualize the analysis results on the web browser by using Hadoop User Experience (Hue). Our result can be helpful for many people.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Computer Networks; D.2.8 [**Software Engineering**]: Metrics—*big-data analysis, distributed systems*

General Terms

Computer Networks, Analysis of Network Packet, Network Security

Keywords

Computer Networks, Packet Data Analysis, Distributed Systems, Network Security

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

1. INTRODUCTION

According to report [5] by International Telecommunication Union (ITU), seven billion people live in an area that is covered by a mobile-cellular network. It means that the Internet is one of most necessary things in human beings' lives. There are many services on the Web such as social network services. As the results, the companies, governments, etc hold many types of private information on servers connected to the Internet. This information include: password, credit card numbers, individual numbers, etc. They should defend the information from crackers who are maliciously attack to steal it. However, there are many news reporting stolen such information. Therefore, analyzing the network packets is one of most important tasks to defend against such cyber attacks.

Our previous works [11, 12] studied the development of an intelligent detection systems against distributed denial of service (DDoS) attacks by analyzing large network packet data that is about 44GB. The results showed that the detection accuracy is high and the results of network packet analysis are very useful to develop detection systems against network attacks such as DDoS attacks.

We analyzed the provided packet capture (pcap) files by using Wireshark [14]. We extracted some features including source IP address, destination IP address, time interval in seconds between packets, and packet size in bytes from the dataset. After that, we analyzed the extracted data. However, it took very long time to finish this process. Therefore, the techniques that can analyze the pcap file directly and visualize the results are required. In this paper, we stored all pcap files to Hive table [2], analyzed the data by executing Hive SQLs, and visualized the results by using Hadoop [1] and Hadoop User Experience (Hue) [4].

The remainder of this paper is organized as follows. In Section 2, we shortly review publications in the area of this research. In Section 3, we explain our experimental environment such as Hadoop, Hive, etc. In Section 4, we present how we can analyze pcap data in the environment and how to visualize the results. In Section 5, we highlight our significant findings.

2. RELATED WORK

Wireshark [14] is open source software and one of most famous analyzing tools for network packets, but there are some limitations. Mistry et al [13] presented four different network monitoring tools that can monitor and analyze the network traffic. They discussed the disadvantage of Wire-

shark. According to them, it will not detect malicious activities on the network and it means that Wireshark may not be useful to study network security. In addition, Wireshark cannot handle the large packet data. According to Wireshark wiki [8], if the user has a large capture file more than 100MB, Wireshark will become slow while loading, filtering and alike actions.

Asrodia and Patel [9] studied the basics of packet sniffer that is used for network traffic analysis. To carry out this analysis, they introduced several tools. There are many tools to capture network traffic and analyze the data. However, there are some limitations for each tool. Some tools may only capture network traffic without analysis or require large memory.

Bachupally et al [10] presented an approach to analyze the network packet data and detect anomalous connections to the network by using the Hadoop Distributed File System (HDFS). They extracted some features of pcap data that size is 131MB by using Wireshark, exported them to the csv file, and uploaded to the HDFS environment. However, they handled small data and also used Wireshark to extract data into the csv file. It means that their approach requires long time for getting results of the analysis.

From these research reviews, we detected that one of the most difficult problem is to handle the large packet data and analyze it. Thus, in this study, we designed the analysis environment to handle the large packet data by using Hadoop, Hive, and Tez.

3. EXPERIMENTAL ENVIRONMENT

3.1 Apache Hadoop

Apache Hadoop is the open source software for reliable, scalable, distributed computing. The Apache Hadoop software library is a framework that allows the distributed processing of large datasets across a cluster of computers using simple programming models. It is designed to scale up from a single server to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures [1]. On Hadoop, the main execution engine is MapReduce. It s the core of Hadoop and it allows massive scalability for a huge Hadoop cluster.

3.2 Apache Hive

Apache Hive is the open source data warehouse software. It facilitates reading, writing, and managing large datasets residing on distributed storage using SQL. A command line tool and JDBC driver are provided to connect users to Hive [2]. Hive supports analysis of large datasets stored in HDFS.

In this study, we stored the provided dataset to the Hive table. However, there is no default option to handle the pcap files in Hive. So, we utilized the Hadoop PCAP library developed by Réseaux IP Européens Network Coordination Centre (RIPE-NCC) [6].

3.3 Hive on Apache Tez

Apache Tez [3] is a new application framework that can execute complex directed acyclic graphs of general data processing tasks and it can be a flexible and powerful successor of the MapReduce framework. MapReduce was initially created for processing and generating large data sets with parallel distributed algorithms on a cluster.

3.4 Hadoop User Experience (Hue)

Hadoop User Experience (Hue) is the open source software and web interface for analyzing data with Apache Hadoop [4]. Hue has editors for Hive, HBase, Spark, etc. In this study, we utilized Hue to execute hive SQL and visualize the results on the web browser.

3.5 Dataset

The data that we utilized in this study were provided by Information Security Centre of Excellence at the University of New Brunswick [7]. This dataset consists of labeled network traces, including full packet payloads in the pcap format for seven days of network activity. The size is about 85GB.

4. TOOLS TO ANALYZE AND VISUALIZE THE DATA

In this study, we set up the standalone mode in Hadoop and executed SQLs on a local machine.

Figure 1 shows the Hive page on the Hue. On the left side of the page, we can select the database and show the tables in the database. In this case, we selected the database named *"default"* and showed the all tables. In the middle of the page, we can write and execute on an Hive SQL and the SQL results are displayed. On this page, we can see the submitted query and the result.

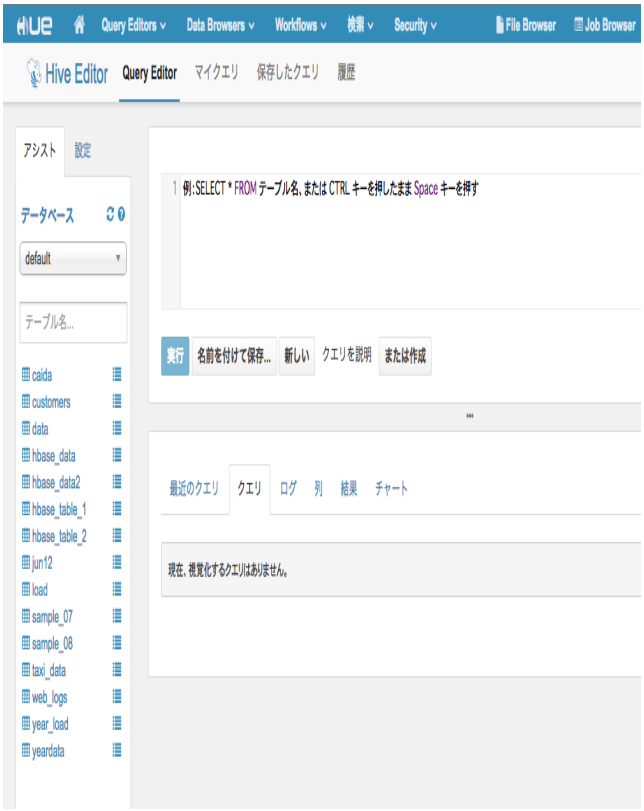


Figure 1: Hive page on Hue

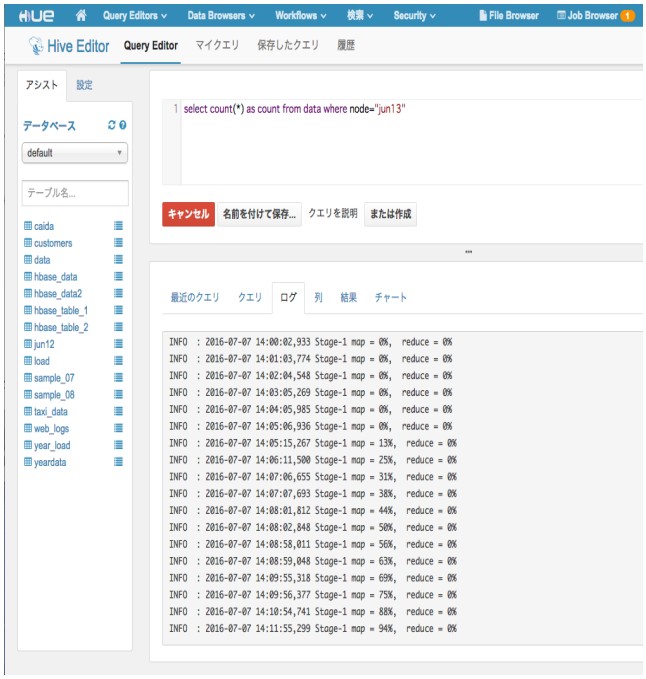


Figure 2: Execution of Hive SQL on MapReduce

Figure 3: Job Browser on Hue

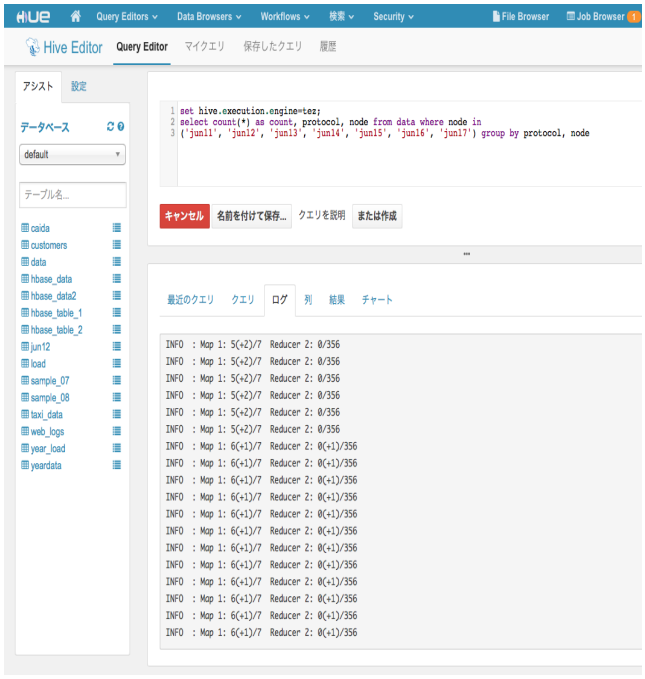


Figure 4: Execution of Hive SQL on Tez

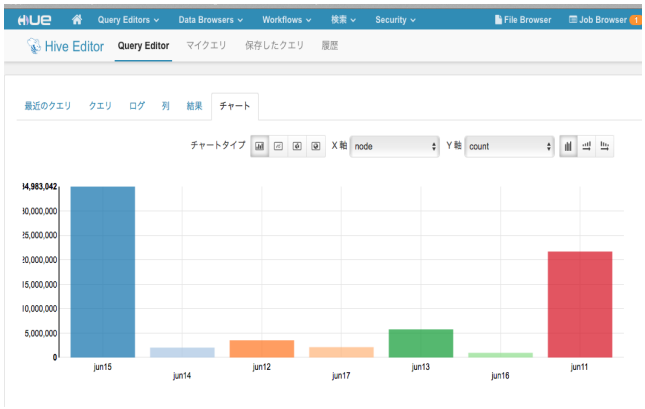


Figure 5: Bar graph of the total number of records for each day

Figure 2 shows an example of the Hive SQL execution to count the number of records in *data* table that is partitioned by *"node"* and node name is *jun13*. In the middle of Figure 2, *INFO: 2016-07-07 14:00:02,933* shows that we executed the SQL at 14:00:02,993 on July 07, 2016 and *Stage-1 map = *%, reduce = *%* shows how percentage of map and reduce job were finished. Figure 3 shows more details on the status of the MapReduce job. On this page, we can see the log by clicking the square, execution ID, the executed SQL, the status of the job, the user name, a percentage of the map and reduce job, a queue, priority (if it sets), the elapsed time, and time when the SQL was submitted from left side. The status is categorized by four colors: Green shows successfully executed, Yellow shows running the SQL, Red shows the failed execution, and Black shows the forced termination.

Figure 4 shows an example of executing Hive SQL on Tez. The first line in the SQL sets the execution engine in Hive from MapReduce to Tez. This SQL counts the number of records for seven partitions in the data table. In comparison, on Figure 2, a past below of the SQLs shows the percentages of the finished map and reduce job. However, in this case, it shows how many jobs of the total job number are finished. In addition, the log shows how many workers are assigned to map and reduce job. In this study, we set that number of workers to two. On the log, *INFO : Map 1: 5(+2)/7 Reducer 2: 0/356* shows that there are total 7 map jobs and 356 reduce jobs, five map jobs are finished and two workers are assigned to finish last two map jobs. *INFO : Map 1: 6(+1)/7 Reducer 2: 0(+1)/356* shows that six map jobs are finished and one worker is assigned to finish last map job and one worker is assigned to finish the reduce job. Using Tez, we can finish executing Hive SQL quickly compared to MapReduce.

To visualize the result of SQLs on Hue, we can select four types of graphs: Bar, Line, Circle, and Map by selecting x

and y axis or latitude and longitude. After finishing the SQL (see Figure 4), we can see the result and visualize the result as shown in Figure 5. We can sort the result by ascending or descending order. In addition, we can save the SQL result to a new Hive table in the database or save it as a csv or excel spread sheet (xls) file and visualize it by using other visualization tools.

5. CONCLUSION

Due to the increase influence of the Internet and web services on the user's lives, many people require intelligent defense systems against cyber attacks. In network security, one of most important problem is to develop an intrusion detection system (IDS), intrusion prevention system (IPS), web application firewall (WAF), etc.

In this paper, we introduced Hadoop, Hive, Tez and designed the analysis environment to handle large packet data. We discussed the advantages of the techniques to analyze pcap files directly and how they are powerful to handle the pcap files. In addition to these techniques, there are many frameworks and tools to analyze many types of data and develop different systems. The development of these techniques is also very active and many libraries are dedicated for these purposes. We can execute some machine learning algorithms and visualize the results by using the machine learning libraries in Apache Spark, Apache Mahout, etc.

As our future work, we will utilize these techniques to analyze the large network packet data in near real-time and apply some machine learning algorithms to develop near real-time automatic detection systems against network attacks.

6. REFERENCES

[1] Apache Hadoop. `http://hadoop.apache.org/`.

[2] Apache Hive. `http://hive.apache.org/`.

[3] Apache Tez. `http://tez.apache.org/`.

[4] Hadoop User Experience. `http://gethue.com/`.

[5] ITU ICT Facts and Figures 2016. `http://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2016.pdf`.

[6] Réseaux IP Européens Network Coordination Centre. `https://www.ripe.net/`.

[7] UNB ISCX Intrusion Detection Evaluation DataSet. `http://www.unb.ca/research/iscx/dataset/iscx-IDS-dataset.html`.

[8] Wireshark Performance. `https://wiki.wireshark.org/Performance`.

[9] P. Asrodia and H. Patel. Network traffic analysis using packet sniffer. *International Journal of Engineering Research and Applications (IJERA)*, Vol 2(Issues 3):854 – 856, 2012.

[10] Y. R. Bachupally, X. Yuan, and K. Roy. Network security analysis using big data technology. *Proceeding of IEEE SoutheastCon 2016*, pages 1 – 4, 2016.

[11] K. Kato and V. Klyuev. An intelligent ddos attack detection system using packet analysis and support vector machine. *International Journal of Intelligent Computing Research (IJICR)*, Volume 5(Issues 3/4 Sep/Dec 2014):478 – 485, 2014.

[12] K. Kato and V. Klyuev. Large-scale network packet analysis for intelligent ddos attack detection development. *Proceeding of the 9th International Conference for Internet Technology and Secured Transactions*, pages 360 – 365, 2014.

[13] D. Mistry, P. Modi, K. Deokule, A. Patel, H. Patki, and O. Abuzaghleh. Network traffic measurement and analysis. *2016 IEEE Long Island Systems, Applications and Technology Conference (LISAT)*, pages 1 – 7, 2016.

[14] Wireshark. `https://www.wireshark.org/`.

Comparison of Bootstrap and W3.CSS

Kentaro Murai
Software Engineering Lab, University of Aizu
Aizu-Wakamatsu City,
Fukushima-ken, 965-8580, Japan
s1210250@u-aizu.ac.jp

Vitaly Klyuev
Software Engineering Lab, University of Aizu
Aizu-Wakamatsu City,
Fukushima-ken, 965-8580, Japan
vkluev@u-aizu.ac.jp

ABSTRACT
Nowadays, there are many device categories used to browse webpages on the Internet. Desktops, laptops, computers, smartphones, and tablets are known examples. Usage of mobile devices drastically changed requirements for the website development. A webpage that can provide an appropriate layout and content for visitors is needed. Bootstrap and W3.CSS are frameworks for modern website development. In this research, we characterize these two frameworks and compare their functionalities.

Categories and Subject Descriptors
H.3.5 [**Online Information Services**]: Web-based services

General Terms
Languages

Keywords
HTML, CSS, Responsive, Grid Layout

1. INTRODUCTION
Bootstrap [1] and W3.CSS [2] are the frameworks that can produce modern webpages. Especially, these frameworks are good at making webpages suited for browsing by devices with displays of different sizes. Webpages with these features are known as responsive webpages. W3.CSS is younger than Bootstrap. Therefore W3.CSS is no well known. As a result of searching on GitHub [3] for each framework name, the keyword "bootstrap" gives 67,439 repository results. On the other hand, the keyword "w3.css" returns just 33 repository results. The search was done in August 2016. These two frameworks are compared in this study.

Studies [4] and [5] suggest the key features of responsive web design. They are:

- Flexible (fluid) grid,
- Flexible images, and
- CSS3 media queries.

ICAIT'16, Oct. 6–8, 2016, Aizu-Wakamatsu, Japan.

In our comparison, we consider these key features and some functions related to them.

2. RELATED WORK
Study [6] surveys current situation of the web development. They summarize main advantages of responsive websites: user-friendly, less maintenance, and no additional domain names.

Studies [4] and [7] mention that responsive web design has an advantage in terms of Search Engine Optimization (SEO). In this research, we also consider the mobile SEO for each framework.

Nowadays, Bootstrap is a main tool to develop responsive websites. W3.CSS is a new instrument for the same purposes. This study is the first attempt to compare these frameworks.

3. FRAMEWORKS
3.1 Bootstrap
Bootstrap is a framework utilizing HTML, CSS, and JavaScript to develop webpages. It was originally created by Twitter in 2010. It is an open source project. Beginning from version 2, it is capable to design responsive webpages. Now, Bootstrap 4 is being developed.

3.2 W3.CSS
W3.CSS is also a framework to develop webpages. In contrast to Bootstrap, W3.CSS doesn't need JavaScript. It uses only HTML and CSS technologies. The first version of W3.CSS was released in May 2015. It is inspired by Google Material Design since its first version.

4. MAIN FEATURES
4.1 Flexible Grid
Bootstrap and W3.CSS support a grid layout (flexible grid). Each framework can divide screen rows up to 12 virtual columns.

The following Bootstrap code divides a row whose width is the window size into 4/12 and 8/12. In this case, the number 4 in the class name col-xs-4 in the third div element indicates four twelfth and the number 8 in the class name col-xs-8 indicates eight twelfth

```
<div class="container-fluid">
    <div class="row">
        <div class="col-xs-4"
                    style="background-color:blue;">
            <p>COLUMN</p>
        </div>

        <div class="col-xs-8"
                    style="background-color:grey;">
            <p>COLUMN</p>
        </div>
    </div>
</div>
```

The following code in W3.CSS produces the same result as the aforementioned code in Bootstrap. In this situation, the class s4 in the second div element represents 4/12 and the class s8 represents 8/12.


```
<div class="w3-row">
    <div class="w3-col s4 w3-blue">
        <p>COLUMN</p>
    </div>

    <div class="w3-col s8 w3-grey">
        <p>COLUMN</p>
    </div>
</div>
```

W3.CSS has another way to divide a row. The following code divides a row into 1/3 and 2/3. The class w3-third in the second div element represents one third and the class w3-rest means the remaining part of the row width: two thirds of the row.

```
<div class="w3-row">
    <div class="w3-container w3-third w3-blue">
        <p>COLUMN</p>
    </div>

    <div class="w3-container w3-rest w3-grey">
        <p>COLUMN</p>
    </div>
</div>
```

4.2 Flexible Images

Flexible images are the function to fit the images into parent elements. If the size of the image is fixed, image is not displayed appropriately on various devices.

The following code is an example of Bootstrap enables the flexible image.

```
<img src="img/picture.jpg" class="img-responsive"
                                alt="Sample Picture">
```

The same functionality in W3.CSS can be implemented as follows:

```
<div class="w3-container">
    <img src="img/picture.jpg" alt="Sample Picture"
                                  style="width:100%">
</div>
```

4.3 CSS3 Media Queries

Media queries are the functions written according to the CSS3 standard. They can vary the layout and contents to display based on the device's width. If the developer changes the layout according to the width without framework, media queries condition should be written. Meanwhile, each framework has concrete media queries condition by default.

The following code is excerpted from W3.CSS.

```
                                ⋮
 1 @media only screen and (min-width:601px){
 2 .w3-col.m1{width:8.33333%}
 3 .w3-col.m2{width:16.66666%}
 4 .w3-col.m3,.w3-quarter{width:24.99999%}
 5 .w3-col.m4,.w3-third{width:33.33333%}
 6 .w3-col.m5{width:41.66666%}
 7 .w3-col.m6,.w3-half{width:49.99999%}
 8 .w3-col.m7{width:58.33333%}
 9 .w3-col.m8,.w3-twothird{width:66.66666%}
10 .w3-col.m9,.w3-threequarter{width:74.99999%}
11 .w3-col.m10{width:83.33333%}
12 .w3-col.m11{width:91.66666%}
13 .w3-col.m12{width:99.99999%}}
14 @media only screen and (min-width:993px){
15 .w3-col.l1{width:8.33333%}
16 .w3-col.l2{width:16.66666%}
                                ⋮
```

Line 1 to 13 mean that when the screen width is between 601px and 992px, each class (e.g. `.w3-col.m1`, `.w3-quarter`) is the specific width.

The grid layout is also used to make a responsive webpage. Screens presented in Figure 1 and 2 are generated from the same code using the grid layout. The code below divides a row into 2/12, 5/12, and 5/12 in W3.CSS. The functionality of grid layout in Bootstrap is almost same as in W3.CSS.

```
<div class="w3-row">
    <div class="w3-col m2 w3-light-grey">
        <p>COLUMN</p>
    </div>

    <div class="w3-col m5 w3-indigo">
        <p>COLUMN</p>
    </div>

    <div class="w3-col m5 w3-red">
        <p>COLUMN</p>
    </div>
</div>
```

Figure 1 simulates a computer's or tablet's wide display (≥ 601px). Figure 2 simulates a smartphone's narrow screen (< 601px).

When the webpage is seen by a narrow width display, the divided columns automatically form lines vertically. This behavior is useful to be seen by any display. The developer doesn't write complex code (e.g. media queries statement) any more to make a responsive webpage.

4.4 Mobile SEO

Mobile SEO [8] is website optimization especially for the mobile users. Google officially recommends responsive web design as the design pattern to optimize a website.

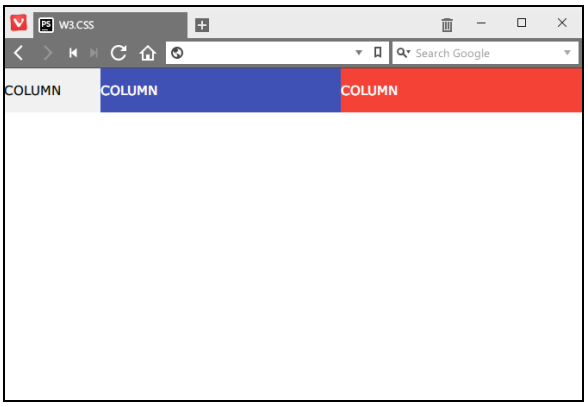


Figure 1. View of the wide display

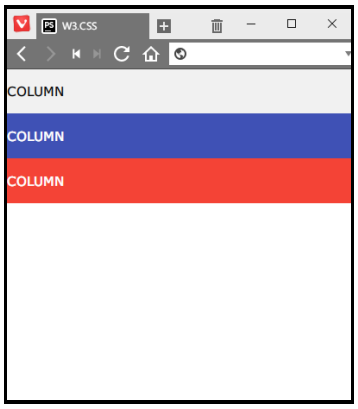


Figure 2. View of the narrow display

Google provides tool [9] to check whether the website is mobile friendly or not. We checked each framework by this tool. The result showed that each framework is regarded as a mobile friendly. Figure 3 shows the result of the test.

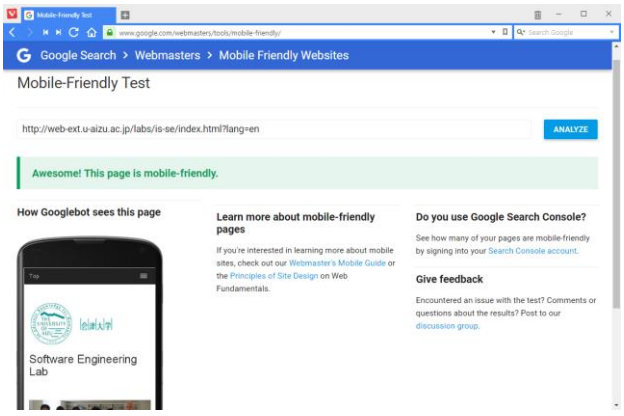


Figure 3. Result of Mobile-Friendly Test

4.5 Initialization

Each framework has the same way to deploy itself to the website. They are as follows:

- Add the link to the officially provided configuration file as link element;
- Download the configuration files to the user's site directly.

Table 1 illustrates characteristics of framework components.

Table 1. Frameworks: Characteristics of the files

	Bootstrap (v3.3.6) [1]	**W3.CSS (v2.7)**
File size	279 KB	28 KB
The number of files	16	1

In addition, Bootstrap requires jQuery to be installed. To use W3.CSS, only one file is needed as described above. It's a CSS style sheet file. We can say W3.CSS is much easier for the developer to arrange compared to Bootstrap.

4.6 Colors

Bootstrap has several classes to color texts or the background of the elements. For example, the class `text-danger` colors the texts in red color meaning dangerous.

```
<p class="text-danger">
    This text is dull red colored.
</p>
```

Likewise, the class `bg-waring` colors the background.

```
<p class="bg-warning">
    This background is dull yellow colored.
</p>
```

W3.CSS has several color classes in a form of `w3-`*`COLOR`*. The color palette like a Google Material Design is adopted. To color some elements, the developer only has to add the name of the color class to the class attribute. The following code is coloring p element.

```
<p class="w3-yellow">This is a paragraph.</p>
```

When the developer colors a text itself, `w3-text-`*`COLOR`* classes are used likewise.

```
<p class="w3-text-red">This text is red colored.</p>
```

4.7 Images

To decor the images, Bootstrap and W3.CSS have almost same functionality. The developer only has to add a specific class to the class attribute. This operation doesn't need time and effort.

[1] Bootstrap is distributed as one zip archive file.

The following code is in Bootstrap.

```
<img src="img/picture.jpg" class="img-rounded"
                                    alt="windmill">
<img src="img/picture.jpg" class="img-circle"
                                    alt="windmill">
```

Figure 4. Rounded and circled images

The following code is in W3.CSS.

```
<img src="img/picture.jpg" class="w3-round"
                                    alt="windmill">
<img src="img/picture.jpg" class="w3-circle"
                                    alt="windmill">
```

The only difference is the class name. Each framework generates the same result.

4.8 Tables

Making a table readable by using Bootstrap or W3.CSS is easy. Table 3 is an example without usage any framework or CSS function. Table 4 is the example of table created in Bootstrap.

Table 2. Example of table in HTML

No	Prefecture	Capital
1	Miyagi	Sendai
2	Yamagata	Yamagata
3	Fukushima	Fukushima

Table 3. Example of table created in Bootstrap

No	Prefecture	Capital
1	Miyagi	Sendai
2	Yamagata	Yamagata
3	Fukushima	Fukushima

To improve visualization of a table (see Table 4), the developer should add the specific class to the table element. When using Bootstrap, this class is as follow:

```
<table class="table table-striped">
```

To do this in W3.CSS, the developer has to write the table element as follow.

```
<table class="w3-table w3-striped">
```

Each framework has other classes to decor the tables.

5. CONCLUSION

Bootstrap and W3.CSS are modern web developing frameworks. Both frameworks provide similar functionality and easy to use to develop a webpage. Because W3.CSS is newer than Bootstrap, W3.CSS has some unique features to make a webpage viewable. Bootstrap is popular among the developers and there are many examples of its utilization in the GitHub repositories. A key advantage of W3.CSS is in the following: It does not need any extra tool for its functionality. It is simple from the point of view of requirements and easy to use.

6. REFERENCES

[1] Bootstrap, Bootstrap, http://getbootstrap.com/ [Accessed on August 22, 2016]

[2] W3Schools, W3.CSS Home, http://www.w3schools.com/w3css/ [Accessed on August 22, 2016]

[3] GitHub, Inc., GitHub, https://github.com/ [Accessed on August 22, 2016]

[4] S. Mohorovičić, "Implementing Responsive Web Design for Enhanced Web Presence", MIPRO 2013, pp.1206-1210, 2013.

[5] Abdulrehman A. Mohamed, Cheruiyot W.K, Richard Rimiru, Collins Ondago, "Responsive Web Design inFluid Grid Concept Literature Survey", The International Journal Of Engineering And Science (IJES), Volume 3, Issue 7, pp.49-57, 2014.

[6] Wei Jiang, Meng Zhang, Bin Zhou, Yujian Jiang, Yingwei Zhang, "Responsive Web Design Mode and Application", 2014 IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA), pp.1303-1306, 2014.

[7] Jari-Pekka Voutilainen, Jaakko Salonen, Tommi Mikkonen, "On the Design of a Responsive User Interface for a Multi-Device Web Service", 2015 2nd ACM International Conference on Mobile Software Engineering and Systems, pp.60-63, 2015.

[8] Google, Mobile SEO Overview, https://developers.google.com/webmasters/mobile-sites/mobile-seo/ [Accessed on August 22, 2016]

[9] Google, Mobile-Friendly Test, https://www.google.com/webmasters/tools/mobile-friendly/ [Accessed on August 22, 2016]

Noise Removal Methods from Web Pages

Vasilii Korelin
Saint-Petersburg State University
7-9 Universitetskaya Naberezhnaya,
St. Petersburg, 199034, Russia
+7 911 266 12 99
vn.korelin@gmail.com

Ivan Blekanov
Saint-Petersburg State University
7-9 Universitetskaya Naberezhnaya,
St. Petersburg, 199034, Russia
+7 921 339 53 43
i.blekanov@gmail.com

ABSTRACT
Almost all the pages of websites of large organizations have a variety of markups, headers, footers and menu items. In terms of information retrieval, this part of the page is not semantically significant for it and can be considered as noise. Furthermore, noise can negatively affect information retrieval results. Therefore, eliminating noisy information is an important step in pre-processing for subsequent analysis (clustering, classification, etc.). This paper discusses several methods of noise removal from Web pages belonging to a large collection. The first method is based on the use of Boilerpipe library to detect and remove surplus "clutter" (boilerplate, templates) around the main textual content of a Web page. The second method is based on a headless browser. The third method involves the use of HTML5 semantic markup (only applicable for browsers that support HTML5). An experiment to assess the quality and performance speed of the methods described is presented. Comparative analysis is carried out.

Categories and Subject Descriptors
H.2.8 [**Database Applications**]: Data mining; H.3.3 [**Information Search and Retrieval**]: Information filtering.

General Terms
Algorithms, Performance, Experimentation, Languages.

Keywords
Noise detection, noise elimination, HTML5, Boilerpipe, headless browser, document preprocessing.

1. INTRODUCTION
For today, in connection with the large number of websites it became necessary to analyze content of Web pages for further information processing (information retrieval, classification, clustering) [1]. Humans can easily distinguish the main page content from various noisy information such as navigation menus, header and footer elements, advertising and other text portions during website examining. This main content can be considered as semantically significant information. Normally, the Web crawlers that gather Web pages eliminate noise extract semantically significant information and store it for further analysis. There is a variety of techniques for noise detection in Web pages [2]. In this paper we consider three approaches of noisy information removal that are based on the following tools: Boilerpipe library, HTML5 semantic markup, Headless browser with Selenium.

This paper is structured as follows. The Section 2 describes methods that can be used for extraction of main textual content. An experiment to assess the quality of the methods described is presented in the Section 3. The Section 4 is devoted to conclusion.

*ICAIT'*16, Oct. 6–8, 2016, Aizu-Wakamatsu, Japan.

2. NOISE REMOVAL TOOLS

2.1 Boilerpipe library
The Boilerpipe library was written by Christian Kohlschutter for Java platform and released under the Apache License 2.0. It is based on the algorithms that provide detection and removal the surplus "clutter" (templates, boilerplate) which is placed around the main textual content of a Web page. Nowadays, the library supplies specific approaches for common task (e.g., extraction of the news articles) and can be easily enlarged for personal problem issues. The paper "Boilerplate Detection using Shallow Text Features" was the basis of the Boilerpipe library that uses and extends algorithms from it [3]. Analysis of small set of shallow text features is the main approach for classifying the particular text elements of a Web page. The main components of the Boilerpipe library are listed below.

HTML parser

The main purpose of the HTML parser is to transform an HTML page into set of text "blocks" that can be considered as an internal text-only document model. The parser is based on the third-party library, CyberNeko [4]. It transforms an HTML document into a TextDocument which consists of one or more TextBlocks. HTML parser can distinguish specific elements such as Script, Option, etc. and ignore them automatically. Each TextBlock contains a text portion from the HTML document and shallow text statistics for it (e.g., words number and words number in the anchor text).

Filters

Every filter is applied to the TextDocument and normally iterates once over all TextBlocks from it. Every individual TextBlock is marked by filter as content or boilerplate. Also filter can assign additional textual label to it. Boilerpipe's filters are grouped as follows:

1. Basic filters

2. English filters, that can be applied to English text (they might also be applied to the other Western languages, but some settings perhaps need to be changed).
3. Heuristics Filters was not explored but will be investigated in the future.

Extractors

Extractors are formed by one or more Filters. Their main purpose is to obtain content from a Web page. For instance, extractor that uses "pipelines" filters takes the parsed HTML document and extracts the main textual content from it. There are several different extractors, from generic DefaultExtractor to special extractors (e.g., ArticleExtractor which is used for news articles).

HTML highlighter

HTML highlighter is an additional tool to represent extracted main content of a Web page as an HTML document.

The algorithms used in Boilerpipe are quite content-independent. However, if the page contains insufficient HTML text (i.e., PDF, Flash or JavaScript) the correctness of the library work is not guaranteed.

2.2 HTML5 Semantic Markup

Such tags as and have little meaning for those who examines an HTML code whether it is machine or human. elements are typically designed for positioning the content on the page. elements are responsible for special formatting of the content. Developers have been using elements to combine page layout, and the developer usually provides the meaning of each element which is based on its id or CSS class.

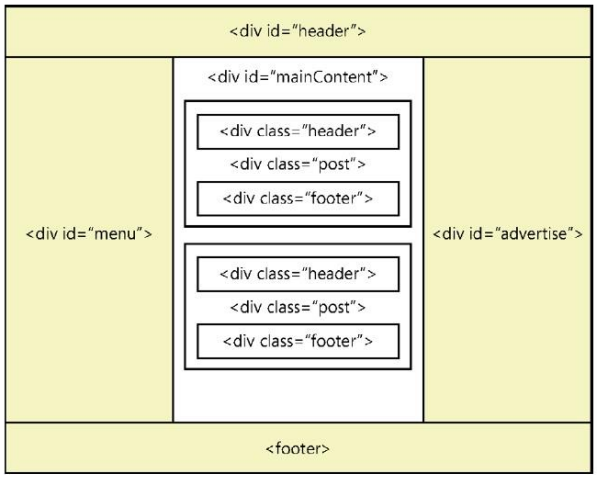


Figure 1. A blog site layout container using elements

HTML5 separates presentation, structure and behavior. Semantic is defined as the study of meaning of linguistic expressions [5]. The HTML5 standard introduces tags that do not serve for presentation purposes but provide meaning. Users usually do not read an HTML code during website surfing, but many machines read it to interpret the Web page. Moreover, Nonvisual Desktop Access (NVDA) devices can provide other means of Web pages processing.

HTML5 semantic elements can be used for creation of layout container which has elements that are meaningful to both the developer and the machine. The following are common elements by which an HTML5 layout container can be created.

- <header> specifies a header section of HTML document and can be used as a page header. Moreover, it can be place at the <article> element.
- <footer> specifies a section that is placed at the bottom of the HTML document. Moreover, it can be placed at the bottom of the <article> tag content.
- <nav> specifies a section that contains a block of links used for navigational purposes.
- <aside> specifies a section that is normally used for sidebars.
- <section> specifies a section that contains <h1> to <h6> internal elements.
- <article> specifies a section that can be considered as a unit of content (e.g., blog posts, news articles).

In Figure 2, all elements have been replaced with the HTML5 semantic elements.

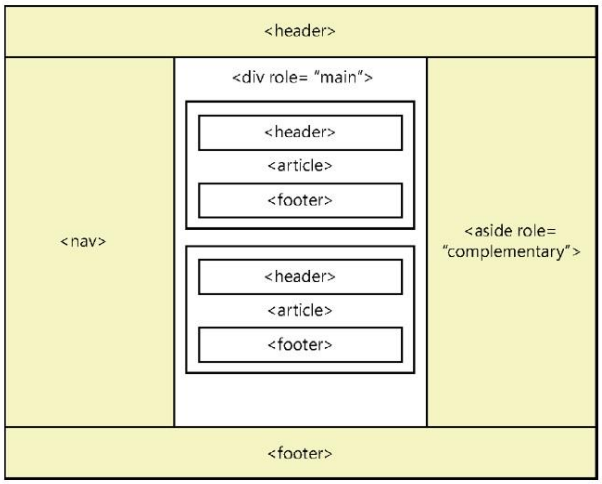


Figure 2. Layout container example, using the new HTML5 elements

Web Accessible Initiative (WAI) specifies the Accessible Rich Internet Applications (ARIA) suite (WAI-ARIA). WAI-ARIA defines a role classes that can used for providing additional meaning for HTML page elements. For instance, screen readers can use it for accessibility purposes [6]. There are several parent and child role classes. "Landmark" class is a parent role class that describes regions of the Web page and can be set as an attribute of the HTML tag. The following are child classes of the "Landmark" role class.

- *Application* defines an area that is declared as a Web application.

- *Banner* defines an area with site-specific content (e.g., site name, logo).
- *Complementary* defines an area for an additional page content that can have different meaning than main textual content.
- *Contentinfo* defines an area that contains information about document (copyright notices, links). It is typically footer content, one per Web page.
- *Form* defines an area that contains input controls sending gathered data to server.
- *Main* defines an area for the main content of Web page.
- *Navigation* defines an area for navigational links.
- *Search* defines an area of input controls for query entering and displaying corresponding information.

An <article> tag can be used for noise removal from Web pages. Furthermore, text that is placed into tags with Role attribute which have an appropriate value (i.e., Main) can be also considered as semantically significant.

2.3 Headless browser

One of the methods of Web site content extraction is based on Headless browser that is a Web browser without a graphical user interface. Usage of the headless browser provides a control a Web page via a command line interface or other network communications in environment that is similar to popular browsers. Using Selenium framework that is designed for testing of Web applications user has access to various elements on the Web page [7]. Today, Selenium does not provide methods of noisy information removal. Without any additional methods Selenium framework can extract only all Web page textual content. Therefore the amount of noisy information can be determined among the all textual content of web page. Also Jaccard similarity for headless browser can be compared with Jaccard similarity for other noise reduction methods and determine if these methods should be used for extraction of semantically important text. For instance, method is not appropriate for extraction semantically significant information in case of Jaccard similarity evaluated for this method is less than Jaccard similarity evaluated for headless browser.

3. METHODS EVALUATION

For comparative analysis of methods considered in Section 2 test documents collection was chosen which contains 30 documents with different design, topics and amount of semantically significant information. In these documents all the noise was removed by expertise (manually) and the main content was extracted. This content can be considered as the expected text in comparison with the text that is obtained using the methods of noise removal. In the current paper Jaccard similarity and Shingles algorithm are used to evaluate the similarity of two texts [8].

3.1 Jaccard similarity

A document represents a string of characters. Shingling is a process that creates sets of k-shingles. Define a k-shingle for a document to be any substring of length k found within the document. For instance, $k = 2$, doc="a b c a b". Then after shingling next sets can be obtained: {a, b}, {b, c}, {c, a}. Similar documents to each other will have a lot of equal shingles [8].

In this paper Jaccard similarity is used to determine the similarity of two documents. The Jaccard similarity of sets C_1 and C_2 is the ratio of the cardinality of the intersection of C_1 and C_2 to the cardinality of their union [9].

$$Sim(C_1, C_2) = \frac{|C_1 \cap C_2|}{|C_1 \cup C_2|}$$

Every document can be considered as a set of shingles where every shingle contains k words. Therefore two documents can be represented as boolean matrix with two columns. Rows correspond to the elements of the universal set which is the set of all k-shingles. Columns correspond to the documents. Value of 1 is in the row E and in the column S if and only if the document that corresponds the column S contains the single that corresponds the row E. Otherwise, it is value of 0. Therefore, documents similarity is the Jaccard similarity of the sets of their shingles.

	C_1	C_2
a	1	1
b	1	0
c	0	1
d	0	0

Figure 3. Boolean matrix representing two documents

$Sim(C_1, C_2) = \frac{1}{3}$ – similarity of two documents.

3.2 Experimental results

Tables 1, 2, and 3 present the Jaccard similarity between the expected main textual content and the text that was extracted by the methods discussed above. Shingles of different lengths (2, 5, 10 and 15 words per shingle) are used. The test set of documents was divided into 3 groups: sites with small (Table 1), middle (Table 2) and large (Table 3) main textual content.

Table 1. Comparison of the noise removal results for small size main textual content

Method	k=2	k=5	k=10	k=15	Average
Boilerpipe library	73%	71%	69%	68%	70.25%
HTML5 Semantic markup	49%	48%	48%	47%	48%
Headless browser + Selenium	24%	23%	22%	21%	22.5%

Table 2. Comparison of the noise removal results for middle size main textual content

Method	k=2	k=5	k=10	k=15	Average
Boilerpipe library	83%	82%	80%	78%	80.75%
HTML5 Semantic markup	78%	76%	73%	71%	74.5%
Headless browser + Selenium	71%	70%	69%	68%	69.5%

Table 3. Comparison of the noise removal results for large size main textual content

Method	k=2	k=5	k=10	k=15	Average
Boilerpipe library	93%	88%	83%	79%	85.75%
HTML5 Semantic markup	87%	83%	78%	75%	80.75%
Headless browser + Selenium	79%	75%	71%	68%	73.25%

Comparative analysis shows that the more semantically significant information is present on the Web page the more Jaccard similarity between the expected main textual content and the text obtained by described methods as the more textual content is on the Web page, the more shingles are in the intersection for both documents. Moreover Jaccard similarity grows as the number of words per shingle (k) decreases because for small values of k there are more shingles in the intersection than for large k value. For large size main textual content and small k value Jaccard similarity is greatest because in large size text the shingle with 2 words is much more common for two documents than the shingle that contains 10 or 15 words. For documents with small and middle size textual content the number of shingles in intersection for $k = 2$ is not much different from number of shingles in intersection for $k = 15$. This behavior is typical for all methods.

The highest Jaccard similarity coefficient was shown by the method that is based on Boilerpipe library. Usage of the HTML5 semantic markup eliminates noisy information with less accuracy and can be applied only for those Web sites that use HTML5 semantic markup. Headless browser with Selenium framework renders the entire Web page but cannot remove noisy information. As expected, Jaccard similarity of the last method is significantly lower than Jaccard similarity of the other methods.

4. CONCLUSIONS

This paper describes three methods of noisy text removal for extraction of semantically significant information from Web pages. These methods were tested on a collection of Web pages that have varying markup and amount of main textual content. Experiment shows that Boilerpipe library removes noisy information with the highest accuracy for all kinds of Web pages.

5. ACKNOWLEDGMENTS

This work was supported by the Russian Foundation for Basic Research, grant № 15-01-06105.

6. REFERENCES

[1] Chakrabarti, S. Mining the Web: Discovering Knowledge from Hypertext Data. Morgan Kaufmann, 2002.

[2] Lan Yi, Bing Liu and Xiaoli Li. Eliminating noisy information in Web pages for data mining. Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, August 24-27, 2003, Washington, D.C.

[3] Christian Kohlschütter, Peter Fankhauser, Wolfgang Nejdl. Boilerplate detection using shallow text features. WSDM'10: Proceedings of the third ACM international conference on Web search and data mining, January 2010.

[4] A. Clark and M. Guillemot. CyberNeko HTML Parser. http://nekohtml.sourceforge.net/

[5] Glenn Johnson. Programming in HTML5 with JavaScript and CSS3. Microsoft Press, A Division of Microsoft Corporation, One Microsoft Way. Redmond, Washington 2013. 205–212

[6] Peter Thiessen, WAI-ARIA live regions and HTML5, Proceedings of the International Cross-Disciplinary Conference on Web Accessibility, March 28-29, 2011, Hyderabad, Andhra Pradesh, India.

[7] Alexander Sirotkin. Web application testing with selenium. Linux Journal, v.2010 n.192, p.3, April 2010.

[8] Vasilii Korelin, Ivan Blekanov, Sergey Sergeev. Clustering of the External WEB Environment of Universities Using a Modified LSH Algorithm. St. Petersburg State Polytechnical University Journal. Computer Science. Telecommunication and Control Systems. Vol. 5, 2015. 79-87.

[9] Jatsada Singthongchai and Suphakit Niwattanakul, "A Method for Measuring Keywords Similarity by Applying Jaccard's, N-Gram and Vector Space," Lecture Notes on Information Theory, Vol.1, No.4, pp. 159-164, Dec. 2013. doi: 10.12720/lnit.1.4.159-164.

Daily Asset Volatility Dynamic Prediction Model, with an Example Using the EUR/USD Currency Pair

Dmitriy Novopashin
Saint-Petersburg State University
7-9, Universitetskaya nab.,
St. Petersburg, 199034, Russia
dmitriy.novopashin@iqoption.com

ABSTRACT
The main objective of this work is to resolve the issue of predicting the daily volatility dynamic based on the history of quotes, while considering aspects of fundamental analysis. The process breaks down into the following stages: developing a theoretical model, implementation of model algorithms, estimating the impact of the news factor.

Categories and Subject Descriptors
D.4.8 [**Operating Systems**]: Performance – *modeling and prediction.*

General Terms
Algorithms, Performance, Economics, Experimentation, Theory.

Keywords
Volatility, financial markets, predictions, forecasts.

1. INTRODUCTION
It is important to make predictions in various applications both in engineering [1] and in economics [2, 3]. The volatility of the currency market is a crucial indicator for traders, because it is the basis for choosing appropriate financial instruments and predicting investment risks. Volatility prediction is one of the most popular approaches to the financial market.

The main concept of this model is to determine the function of the daily volatility dynamic, which is highly independent of economic events and is a certain pattern of volatility fluctuations. After that it is necessary to consider the impact of upcoming news and events.

2. PREPARING INPUT DATA
Input data is based on the history of quotes that can be found, for example, at http://histdata.com/ [4], which are the basis of volatility calculations for a certain period.

For convenience we will divide the data into financial days so that the beginning of the financial day coincides with the opening of the Wellington exchange and the end of the financial day coincides with the closing of the New York exchange. Calculating average volatility through the running window method.

ICAIT'16, Oct. 6–8, 2016, Aizu-Wakamatsu, Japan.

3. MODEL
We will consider the financial day as eight three-hour periods. This is the longest discretization period of the day, where the demarcations coincide with the time of opening and closing of the world's main foreign exchange markets.

Thus, any trade session always includes a strictly even number of segments of the financial day. Assuming that every foreign exchange market influences the volatility dynamic, this approach allows each segment to maintain homogeneity in relation to the number of active foreign exchange markets.

The financial market is a rather delicate instrument that may react not only to important economic and political events, but to essentially any event that influences a large number of people. Therefore the next important step is filtrating input data. It is necessary to exclude time periods that contain abnormal values and statistical random omissions.

The following stage is approximating the volatility function within each time segment. This allows for estimating the volatility trend within the segment.

After that we need to aggregate the results. However, analyzing data gathered from each segment is an insufficient approach. The behavior of the financial market on Monday for example may be dramatically different from that on Friday; a great deal of economic news comes out on certain days of the month, and many contracts expire on certain days of the week or of the month. Therefore results are aggregated not just by time segments, but by days of the week as well.

In this way, the eight time segments of the day, each illustrating the pattern of volatility fluctuations under the influence of the global market at a certain moment, are united into a single financial day pattern.

For a financial day without any significant events the prediction would be as follows (see Figure 1).

At the same time the basic volatility pattern for a saturated financial day doesn't reflect the dynamics of real behavior (see Figure 2).

4. NEWS FACTOR
With rare exceptions, precise dates of important news releases are known in advance, as well as numerous experts' predictions

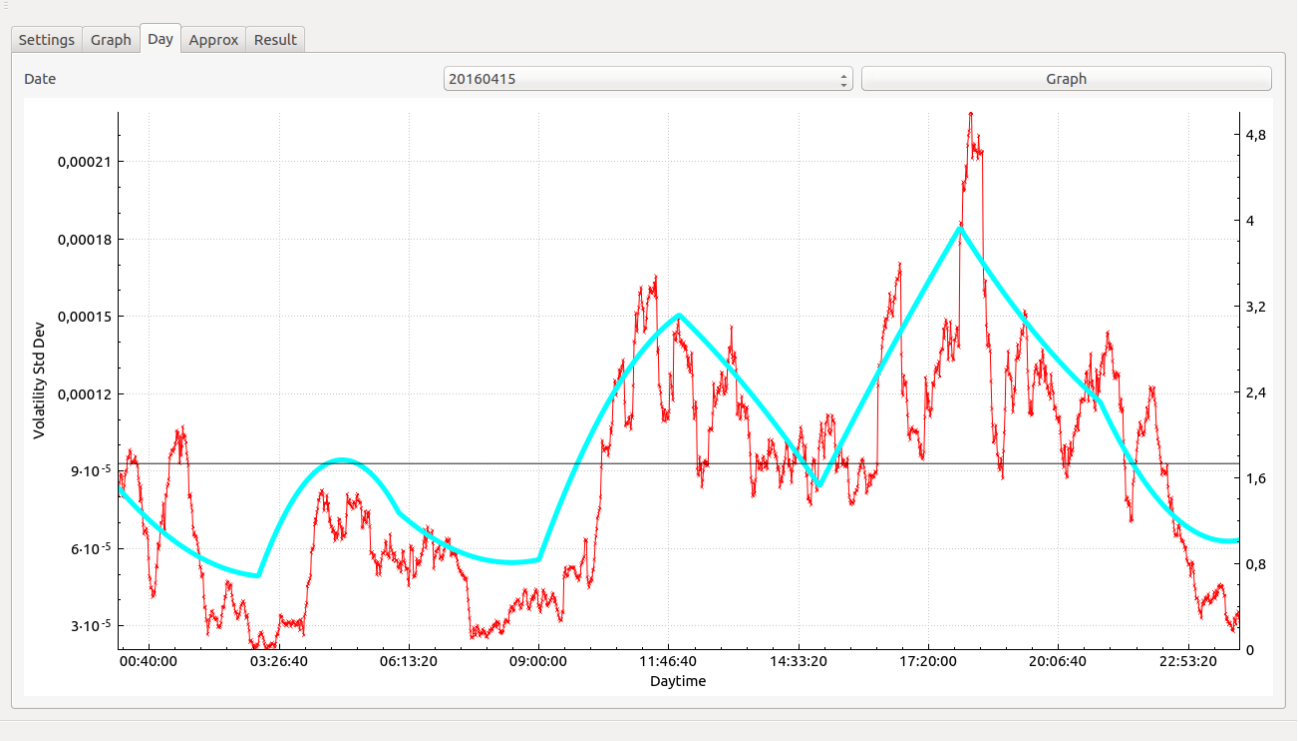


Figure 1. Financial day without news pattern – 4.15.2016. Vertical axis – Std Dev of volatility, horizontal axis – time of day, red curve line – real volatility, black straight line – median, blue line – predicted volatility.

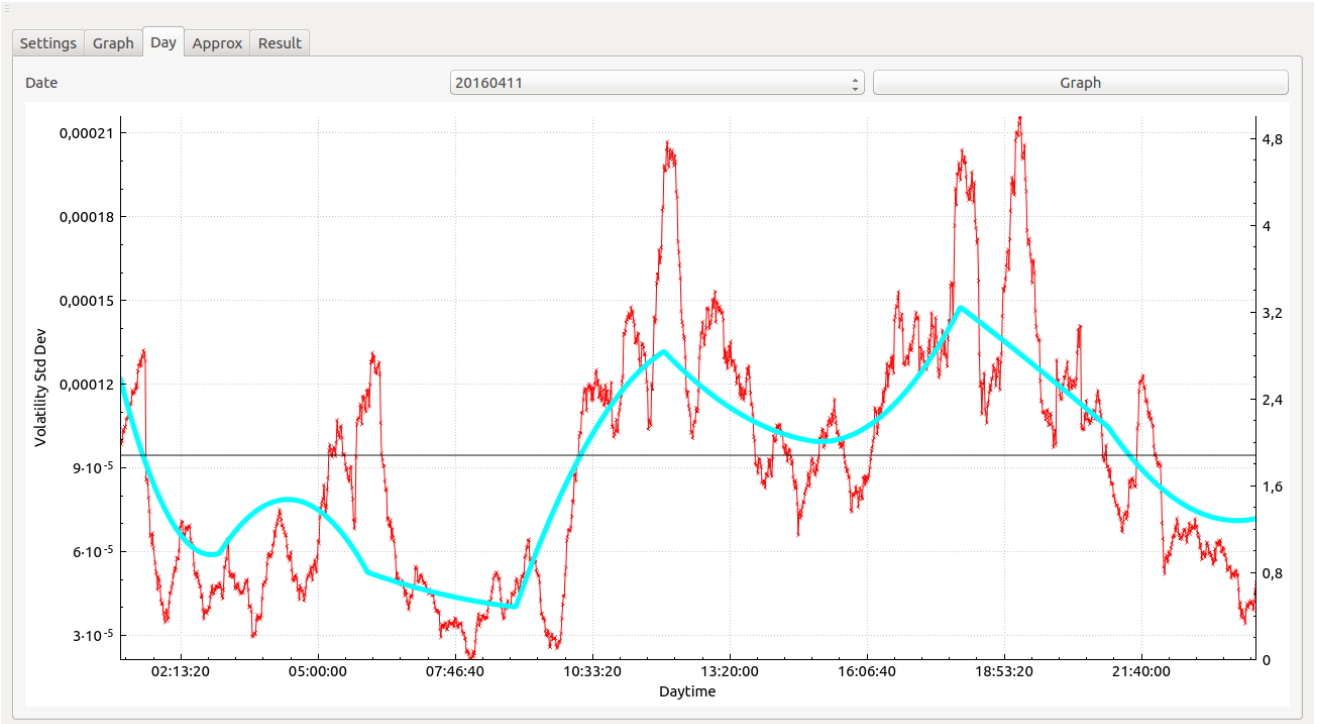


Figure 2. Financial day excluding news pattern – 4.11.2016. Vertical axis – Std Dev of volatility, horizontal axis – time of day, red curve line – real volatility, black straight line – median, blue line – predicted volatility.

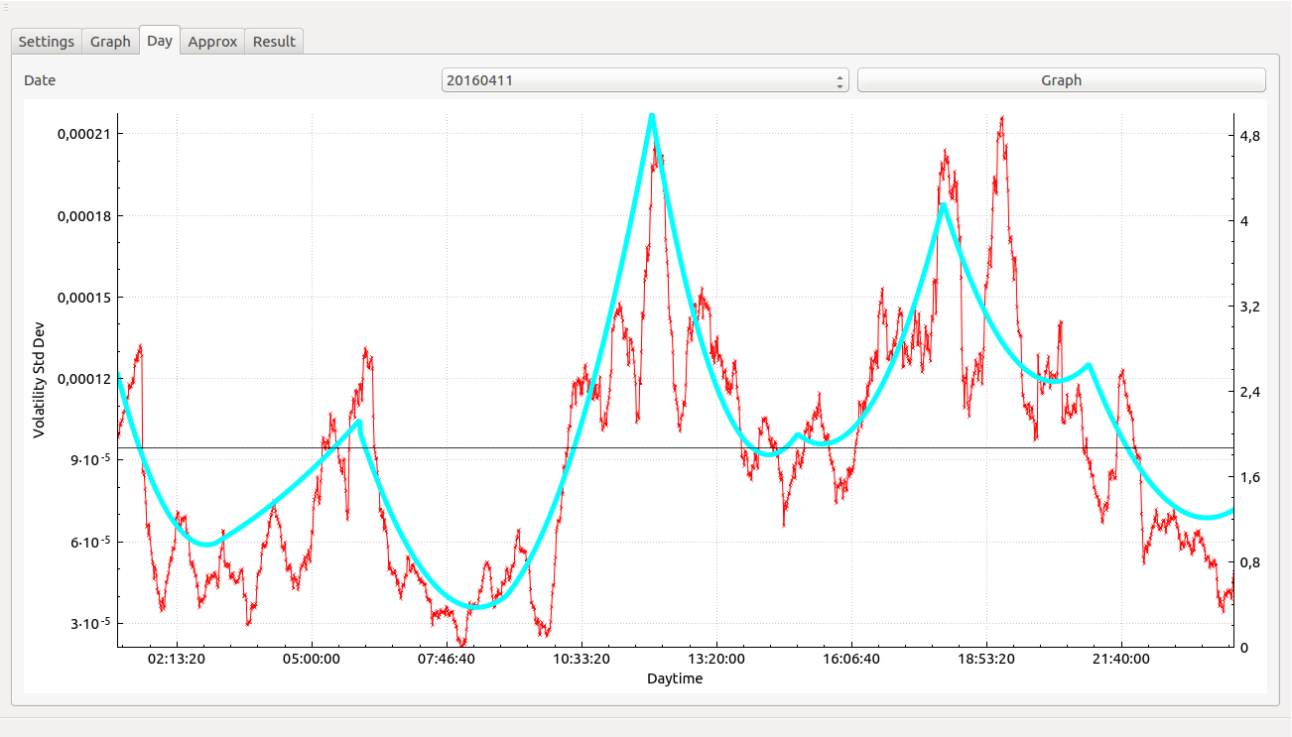


Figure 3. Financial day based on news pattern – 4.11.2016. Vertical axis – Std Dev of volatility, horizontal axis – time of day, red curve line – real volatility, black straight line – median, blue line – predicted volatility.

of their influence. This is also applicable to regular and irregular news. The only difference is the method of estimating their impact on the model. In order to estimate the influence of regular news we can always refer to historic data; for irregular news the only way to interpret it is with expert reviews.

Applying the news quotients will significantly improve the prediction (see Figure 3).

5. PROSPECTS

The prospects of developing and improving the model can be subdivided into two categories. The first category is fundamental statistical estimation of the influence of the news factor, which includes analysis of historic data, accumulating and structuring information as well as examining long-term trends that influence the changes in technical levels of volatility.

The second category is experiments with volatility approximation methods within time segments of the financial day, as well as experiments with the duration of the discreteness of the day, which may be necessary in order to apply the model for other currency pairs and assets.

6. REFERENCES

[1] Novopashin D.V. 2014. Statistical data processing of the secondary radiolocation using orthogonal Chebyshev polynomials. Control Processes and Stability. Vol. 1, No. 1, 352-356.

[2] Prasolov A. V., Zamuraev K. A. 2014. On guaranteed forecast estimation. Vestnik of St. Petersburg University. Ser. 10: Applied Mathematics. Informatics. Control Processes. (4): 82-96, December 2014.

[3] Peresada V. P., Smirnov N. V., Smirnova T. E. 2014. On guaranteed forecast estimation. Vestnik of St. Petersburg University. Ser. 10: Applied Mathematics. Informatics. Control Processes. (4): 119-132, December 2014.

[4] Free Forex Historical Data. URL: http://www.histdata.com/.

An Algorithm to Approximate the Total Number of Site Pages Using a Portion of Its Structure

Sergei Sergeev
Saint-Petersburg State University
7-9 Universitetskaya Naberezhnaya,
St. Petersburg, 199034, Russia
+7 921 381 23 62
slsergeev@yandex.ru

Ivan Blekanov
Saint-Petersburg State University
7-9 Universitetskaya Naberezhnaya,
St. Petersburg, 199034, Russia
+7 921 339 53 43
i.blekanov@gmail.com

Aleksei Maksimov
Saint-Petersburg State University
7-9 Universitetskaya Naberezhnaya,
St. Petersburg, 199034, Russia
+7 911 212 76 76
nightsnaker@gmail.com

ABSTRACT
This article considers the algorithm that allows to measure the webpages and hyperlinks number after the fractional inspection based on the set of equations that correlates the web-crawled and found webpages number. The experiment results that verify algorithm's performance capability were pointed out within.

Categories and Subject Descriptors
G.2.2 [Discrete Mathematics]: Graph Theory – graph algorithms.

General Terms
Measurement, Experimentation.

Keywords
Web-site, hyperlinked structure of the site, webometrics, web-graph, site size determination.

1. INTRODUCTION
One of the main tasks of webometrics is the massive websites inspection to be split into two – deep webpages analysis and its' interaction with web-space [1]. This article generally considers the websites' different attributes as inherent properties. The website structure is commonly accepted (i.e. [2]) to be presented as a direct graph therefore it contains the vertices as webpages, the edges as hyperlinks, connectivity, length etc. In the second place – the ergonomic factors such as website design and usability. Third is the content attributes – the main topic, tags etc. Considering these factors is the key for the preset website selection. The evaluation of these factors is possible after inspecting, for example, the one tenth of the total number of website hyperlinks (that is comparable to the proper book selection – in order to understand the book is worth reading you are to look through the part of it). But to understand what part of the website was inspected already we must possess the number of the webpages (graph vertices). Nowadays there is only one known way to determine the true number of the webpages – the whole web-graph inspection. As long as the massive organizations' websites (i.e. universities) been estimated of tens or even hundreds of webpages with the zillions of hyperlinks its measurement is widely considered to be the resource-intensive task.

The main idea of the offered approximate webpages and hyperlinks calculation method premised on its part is as follows. At first all the hyperlinks on every webpage are to be retrieved. Then every page followed by every hyperlink is to be inspected for another set of hyperlinks etc. Some links lead to new pages, some of them are cycled in order the new-leading links share to be reduced. The relevant logic is in the works [3, 4]. Here the authors put forward the algorithm that allows to measure the website size after the inspection of its part. It is based on the new leading hyperlinks appearing slowdown.

*ICAIT'*16, Oct. 6–8, 2016, Aizu-Wakamatsu, Japan.

2. THE WEBPAGE AND HYPERLINKS NUMBER CALCULATION METHOD

2.1 Mathematical model
Let us consider the approximate procedure of a web-graph random walk.

1. Making the sets:
   a) An *NLinks* $\{a_i\}$. a_i set is the number of retrieved hyperlinks in *i* steps;
   b) An *NPage* $\{v_i\}$. v_i set is the number of webpages that retrieved hyperlinks in *i* steps lead to;
   c) An *UrlsList* set is the number of webpages' domain names that retrieved hyperlinks in *i* steps lead to.
2. Let us retrieve Δa_i hyperlinks from the website and add $a_i = a_{i-1} + \Delta a_i$ to *NLinks* set. Every retrieved hyperlink is to be collated with the *UrlsList* set. If there is no webpage that hyperlink leads to then the hyperlink joins the *UrlsList* set. The added webpages number is Δv_i. Then $v_i = v_{i-1} + \Delta v_i$ been added to the *NPage* set.

Let us denote all the website hyperlinks and webpages number by a_{total} and v_{total} respectively; the number of retrieved hyperlink by a, and the unique webpages number by v. Let us also introduce:

$$a_{remain} = a_{total} - a \text{ and } v_{remain} = v_{total} - v.$$

Let us denote the unique website pages set by V_{total} and still unretrieved webpages number by V_{remain}. Let us break the webpages set into subsets by the incoming hyperlinks. Let us denote the set of the webpages with n incoming hyperlinks by S_n^0 and the subset of webpages within V_{remain}, with n incoming hyperlinks by $S_n(v_{remain})$.

Then $s_n^0 = |S_n^0|$, $s_n(v_{remain}) = |S_n(v_{remain})|$.
The probability of the next retrieved hyperlink leads to the webpage from the $S_n(v_{remain})$ set equals to:

$$p_n(v_{remain}) = \frac{nS_n(v_{remain})}{a_{remain}} = \frac{nS_n(v_{remain})}{a_{total} - a}$$

Let any step has da as the number of hyperlinks retrieved. Then the expected value of the increment $s_n(v_{remain})$:

$$ds_n(v_{remain}) = -\frac{ns_n(v_{remain})}{a_{total} - a}da$$

Let us integrate between 0 and a. As far as $v_{remain} = v_{total}$ with $a = 0$, then:

$$\ln s_n(v_{remain}) - \ln s_n^0 = n(\ln(a_{total} - a) - \ln a_{total})$$

Therefore:

$$s_n(v_{remain}) = s_n^0\left(\frac{a_{total} - a}{a_{total}}\right)^n$$

Let us use the trivial par:

$$\sum_n s_n(v_{remain}) = v_{remain} = v_{total} - v$$

Therefore:

$$v = v_{total} - \sum_n s_n^0\left(\frac{a_{total}-a}{a_{total}}\right)^n \quad (1)$$

It is known (i.e. [5, 6]) that in large networks nodes are distributed by the incoming hyperlinks number follows the law

$$s_n^0 = \frac{\gamma v_{total}}{n^q} \quad (2),$$

with $\frac{1}{\gamma} = \sum_n \frac{1}{n^q}$, and q as an invariable.

$$2<q<3 \quad (3)$$

Let us plug the formula (2) into the equation (1). Then the equation that shows the correlation of the websites pages number v_{total} and the website hyperlinks number a_{total} is as follows:

$$v = v_{total}\left(1 - \gamma\sum_n \frac{1}{n^q}\left(\frac{a_{total}-a}{a_{total}}\right)^n\right) \quad (4)$$

Let us perform the *k* of consequent steps of hyperlinks and related webpages (the pages these hyperlinks lead to) accidental samplings. Let a_i hyperlinks and v_i related webpages were retrieved after the *i*-step. Then the unique equation will correspond to every step. Therefore there is the set of equations that connects the retrieved hyperlinks with the related webpages total numbers v_{total} and a_{total} respectively:

$$v_i = v_{total}\left(1 - \gamma\sum_n \frac{1}{n^q}\left(\frac{a_{total}-a_i}{a_{total}}\right)^n\right), (i = 1,2,\dots,k) \quad (5)$$

2.2 The approximate solution

As far as we need to get the approximate measurement let us assume

$$\frac{1}{n^q} \approx \frac{1}{n(n+1)} \quad (6)$$

Then the limits of this distribution were defined by the inequation (3) (except *n=1*). Therefore the normalizing factor γ:

$$\frac{1}{\gamma} = \sum_n \frac{1}{n(n+1)} = 1 \quad (7)$$

Let us plug the formula (6) into the equation (5) considering (7). Therefore:

$$v_i = v_{total}\left(1 - \sum\frac{1}{n(n+1)}\left(1 - \frac{a_i}{a_{total}}\right)^n\right)$$

Let us analyze the sum

$$\sigma = \sum\frac{1}{n(n+1)}x^n$$

$$\sigma = \sum_1^N \frac{x^n}{n} - \frac{1}{x}\sum_1^N \frac{x^{n+1}}{n+1} = \sum_1^N \frac{x^n}{n} - \frac{1}{x}\sum_2^{N+1}\frac{x^n}{n}$$

As far as the massive websites *N* value reaches several thousand therefore the approximate value:

$$\sigma = -\ln(1-x) + \frac{1}{x}(\ln(1-x) + 1)$$

Therefore:

$$v_i = v_{total}\frac{\frac{a_i}{a_{total}}}{1 - \frac{a_i}{a_{total}}}\ln\frac{a_{total}}{a_i}$$

As far as situation $a_i \ll a_{total}$ is considered let us assume:

$$v = v_{total}\frac{a}{a_{total}}(\ln a_{total} - \ln a)$$

Let us set

$$\frac{a_{total}}{v_{total}} \equiv x, \qquad \ln a_{total} \equiv y$$

Therefore the equation:

$$v_i x + a_i y = -c_i, \qquad \text{где } c_i \equiv a_i \ln a_i$$

Let us solve the overspecified set of equations with the LS method. Therefore:

$$A_{11}x + A_{12}y = C_1$$
$$A_{21}x + A_{22}y = C_2,$$

with

$$A_{11} = \sum_1^k v_i^2\,,\ A_{12} = A_{21} = -\sum_1^k a_i v_i\,,\quad A_{22} = \sum_1^k a_i^2\,,$$
$$C_1 = -\sum_1^k v_i c_i\,,\ C_2 = \sum_1^k a_i c_i \quad (8)$$

Therefore

$$x = \frac{C_1A_{22} - C_2A_{12}}{A_{11}A_{22} - A_{12}^2}\,,\ y = \frac{C_2A_{11} - C_1A_{12}}{A_{11}A_{22} - A_{12}^2} \quad (9)$$

And finally

$$a^* = e^y\,,\ v^* = \frac{e^y}{x} \quad (10),$$

with a^* and v^* are approximate values of a_{total} and v_{total}.
The formulae (8)-(10) allow to approximately measure the website size. According to the assumptions made, the formula can be used only with low value of a_i.

2.3 The experiment

Authors were to perform the experiment in order to test the offered method capability using 11 universities' sites (table 1) and determine the share of the website hyperlinks number needed to measure it approximately (the total number of hyperlinks and the total number of webpages).

Table 1. Universities' sites inspected.

#	University	URL
1	Cambridge	www.cam.ac.uk
2	MIT	www.web.mit.edu
3	Oxford	www.ox.ac.uk
4	The University of Turin	www.unito.it
5	Cornell University	www.cornell.edu
6	Emory University	www.emory.edu
7	Berkeley	www.berkeley.edu
8	Pierre and Marie Curie	www.upmc.fr
9	The University of Aizu	www.u-aizu.ac.jp
10	Syracuse University	www.syr.edu
11	Penn State University	www.psu.edu

Using a web-crawler all webpages and internal hyperlinks have been collected for the each university (from table 1). Actual values of v_{total} and a_{total} were found. Since the offered method proposed is based on accidental samplings, a Fisher-Yates shuffle algorithm described in [7] has been used in order

to shuffle the every website hyperlinks set. Then a hundredth part of hyperlinks was to be picked from the shuffled set of hyperlinks, that made up v_1 and a_1. Then in order to get v_2 and a_2 two hundredth parts of links were to be picked, three hundredth parts for v_3 and a_3 and so on. The obtained values v_i and a_i had been substituted into (4) to calculate a^* and v^*. The proportion of the visited hyperlinks was approximately determined as the ratio of a_i to a^*.

2.4 The experiment results

The table 2 reports the main experiment results.

Table 2. Actual and calculated numbers of webpages and hyperlinks of universities' sites.

University	a_{total}	a^*	v_{total}	v^*	$\frac{a_k}{a_{total}}$
Cambridge	2258483	2993431	69661	114226	0,12
MIT	590474	1363978	70751	116108	0,2
Oxford	1197127	2649532	26289	60634	0,2
The University of Turin	1195984	2975332	22796	32168	0,03
Cornell University	874555	205028	20908	22760	0,03
Emory University	190119	124999	12497	13237	0,07
Berkeley	116240	253220	10544	13207	0,03
Pierre and Marie Curie	548844	154885	9142	6989	0,03
The University of Aizu	116098	78238	3993	4375	0,07
Syracuse University	184773	454720	4979	7576	0,04
Penn State University	32984	78880	1923	3958	0,26
a_{total}, v_{total} – website hyperlinks and webpages number respectively; a^*, v^* - approximate values of $a_{total}$and v_{total}; a_k – the number of hyperlinks used for the calculation					

In table 2 the values a^* and v^* have been obtained by ratio of a_k to a^*, which is equal to 0.1 that based on 2 factors. At first all of the formulae have been obtained using the assumption that $\frac{a_k}{a^*} \ll 1$. The second is if the a_i value is too low then the representativeness of the sample decreases.

Since actual values of the total hyperlinks number differ from the calculated ones, the table shows the actual share of visited links for each university website, for which theoretical values a^* and v^* have been obtained.

3. CONCLUSION

The main results of the work are:

1) The set of equations have been obtained that connects the number of web-crawled website hyperlinks with the webpages being found.
2) There was the experiment performed in order to check the possibility to measure the webpages number premised on its part using the total numbers of the fully inspected webpages and hyperlinks of 11 universities' websites. The experiment proves the worked-out website size measurement method capability after the fractional inspection. Both overestimated and underestimated assumptions (about website size) have been noticed. The relation between the approximately calculated website size value and actual webpages total number is 2.4, and 4.3 – for the hyperlinks if worst comes to worst. The results of the experiment show that there were needed to inspect 3-26% of the total hyperlinks number to approximately measure every university website size. According to the universities' websites information it is apparent that the visited websites' pages differ by 37 times and the visited hyperlinks by 68 times. Therefore the webpages and hyperlinks numbers measurement errors in 2.4 and 4.3 times respectively are considered to be acceptable. The (5) set of equation solution method and the measurement may be improved.
3) The algorithm that implements the offered method was specified. It allows to reduce the costs of approximate webpages and hyperlinks number as well as web-graph hyperlinks measurements significantly.

4. ACKNOWLEDGMENTS

This work was supported by the Russian Foundation for Basic Research, grant № 15-01-06105.

5. REFERENCES

[1] Thelwall, M. 2013. *Webometrics and Social Web Research Methods.* University of Wolverhampton, 8-39.

[2] Broder, A., Kumar, R., Maghoul, F., Raghavan, P., Rajagopalan, S., Stata R., Tomkins, A. and Wiener, J. 2000. *Graph structure in the Web: Experiments and models.* In WWW9, Elsevier Science, 309–320.

[3] Blekanov, I., Sergeev, S. and Klemeshov, E. 2015. *Study of patterns in the hyperlink structure of large sites*. Proceedings of the International Workshop on Applications in Information Technology (IWAIT-2015). The University of Aizu Press, 58-60.

[4] Blekanov, I.S., Sergeev, S. L. and Maksimov, A. I. 2016. *A study of structural characteristics of large websites.* Vestnik of St. Petersburg State University. Series 10. Issue 1, 78-84.

[5] Barabasi, A. and Albert, R. 1999. *Emergence of scaling in random networks. Science*, Vol. 286: 509-512.

[6] Kumar, R., Raghavan, P., Rajagopalan, S., and Tomkins, A. 1999. *Trawling the Web for emerging cyber-communities.* WWW '99 Proceedings of the 8th international conference on World Wide Web: 1481-1493.

[7] Fisher, R., Yates, F. 1948. *Statistical tables for biological, agricultural and medical research (3rd ed.)*. London, Oliver & Boyd, 26–27.

Spectral H_2 Synthesis of Fault Estimation Observer

Evgeny I. Veremey
St. Petersburg State University,
7/9 Universitetskaya nab.,
St. Petersburg, 199034, Russia
Email: e_veremey@mail.ru

Yaroslav V. Knyazkin
St. Petersburg State University,
7/9 Universitetskaya nab.,
St. Petersburg, 199034, Russia
Email: yaroslavknyazkin@gmail.com

ABSTRACT
This paper is devoted to the novel 2-step approach to construction of the fault estimation observer. First, we propose to design the auxiliary fault detection observer, and then to use this one for the initial problem solution. Special spectral algorithm of SISO (Single Input and Single Output) H_2 optimization is implemented in the range of proposed approach. Simulation results are presented to demonstrate applicability and effectiveness of proposed techniques.

Categories and Subject Descriptors
I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search –*control theory*

General Terms
Algorithms.

Keywords
fault detection, fault estimation, control system, optimization, H_2-control, feedback, functional.

1. INTRODUCTION
Fault detection and estimation problems have received serious attention in scientific publications during last two decades (some significant papers are cited in [1-3]). On the other hand, such problems with the initially known spectral features of external disturbances are not explored nowadays.

The matter of this research is to design an adaptive observer for detection and estimation of constant (or slow-varying) additive faults. The technique proposed in this paper is inspired by the special spectral approach to SISO H_2 mean-square optimal synthesis problem [4].

This paper is organized as follows. In Section 2, we present the description and statement of the problem. Sections 3 and 4 describe a design of the auxiliary fault detection observer (the first step) and the required fault estimation one (final step). In Section 5, one numerical example is given to illustrate an implementation of the proposed approach. Section 6 concludes the paper by overall resume.

ICAIT'16, Oct. 6–8, 2016, Aizu-Wakamatsu, Japan.

2. PROBLEM STATEMENT
Let us consider the following LTI system

$$\begin{aligned} \dot{x} &= \mathbf{A}x + \mathbf{b}u + \mathbf{E}f + \mathbf{P}d, \\ y &= \mathbf{C}x, \end{aligned} \tag{1}$$

where $x \in \mathbf{R}^n$ is the vector of the system state, *y, u, f, d* are the following scalars: the output signal, the control one, additive fault action and the external disturbance with the known spectral power density. Let us suppose that the value of the fault derivative may be neglected, i.e. $\dot{f}(t) \approx 0$.

Adaptive fault estimation observer-filter, close to the variant described in [5], has the following structure:

$$\begin{aligned} &\dot{\hat{x}} = \mathbf{A}\hat{x} + \mathbf{b}u + \mathbf{E}\hat{f} + \mathbf{L}_x\nu_1, \\ &\dot{\hat{f}} = \nu_2, \nu_1(s) = V(s)(y - \mathbf{C}\hat{x}), \\ &\nu_2(s) = W(s)(y - \mathbf{C}\hat{x}), \end{aligned} \tag{2}$$

where ν_1, ν_2 are corrective terms, vector $\mathbf{L}_x$, and transfer functions $V(s) = V_1(s)/V_2(s)$, $W(s) = W_1(s)/W_2(s)$ are to be computed. Designed observer-filter (2) should generate fast and accurate fault estimation signal despite of presence of external disturbances.

Let us denote

$$e_x = x - \hat{x},\ e_y = y - \mathbf{C}\hat{x},\ e_f = f - \hat{f}, \tag{3}$$

describe an error dynamics of the observation by the model of the form

$$\begin{aligned} &\dot{e}_x = \mathbf{A}e_x + Ee_f - \mathbf{L}_x\nu_1, \\ &\dot{e}_f = -\nu_2, \nu_1(s) = V(s)e_y, \\ &\nu_2(s) = W(s)e_y. \end{aligned} \tag{4}$$

Rewrite this model in frequency domain:

$$\begin{aligned} sA(s)V_2(s)e_y(s) = -sL(s)V_1(s) - E(s)V_2(s)\nu_2(s) + \\ + sP(s)V_2(s)d(s), \text{ where} \end{aligned} \tag{5}$$

$$\begin{aligned} &A(s) = \det(\mathbf{I}s - \mathbf{A}), E(s) = A(s)\mathbf{C}(\mathbf{I}s - \mathbf{A})^{-1}\mathbf{E}, \\ &P(s) = A(s)\mathbf{C}(\mathbf{I}s - \mathbf{A})^{-1}\mathbf{P}, L(s) = A(s)\mathbf{C}(\mathbf{I}s - \mathbf{A})^{-1}\mathbf{L}_x, \end{aligned} \tag{6}$$

and consider the transfer function from $d(t)$ to $e_f(t)$

$$e_f(s) = F_{e_f d}(s)d(s) = \frac{P(s)V_2(s)W_1(s)}{\Delta_2(s)}d(s), \text{ where} \quad (7)$$

$$\Delta_2(s) = sA(s)V_2(s)W_2(s) + sL(s)V_1(s)W_2(s) + E(s)V_2(s)W_1(s). \quad (8)$$

We suppose that the mentioned disturbances have the following initially given spectral power density

$S_d(s) \equiv S_1(s)S_1(-s)$, $S_1(s) \equiv N_d(s)/T(s)$, where

$$S_1(s) \equiv \frac{N_d(s)}{T(s)} = \sqrt{\frac{4D_r\alpha}{\pi}}\frac{1}{s^2 + 2\alpha s + \alpha^s + \beta^2}, \quad (9)$$

where β is the central frequency, and $\alpha = s_t\beta$, where s_t is the blurriness. Let note that constant factor can be ignored for the SISO case in a range of design algorithms (i.e. $N_d(s) \equiv 1$ in this paper). To simplify disturbance representation, we can also use its polyharmonical form as follows:

$$d(t) = \sum_{i=1}^{n_h} A_{di}\sin(\sigma_i t + \varphi_i) \quad (10)$$

where A_{di}, σ_i, φ_i are amplitudes, frequencies and phases of the corresponding harmonics. The influence of $d(t)$ to the fault estimation process can be expressed by the value

$$J_\omega = \max_i \left\{A_{di}\left|F_{e_f d}(j\sigma_i)\right|\right\}. \quad (11)$$

Remark that this value and the value of the fault estimation process settling time T_p are functions of $\mathbf{L}_x$, $V(s)$, and $W(s)$. In such a way, the problem to be considered is to design the mentioned items of observer such that

$$J_\omega(L,V(s),W(s)) \le J_\omega^0,\ T_p(L,V(s),W(s)) \le T_p^0, \quad (12)$$

where J_ω^0, T_p^0 are given desired values of J_ω and T_p.

3. FAULT DETECTION

First, let us design the fault detection observer-filter, which has the only task: to detect the presence of the fault. The correspondent equations of the filter are as follows:

$$\begin{aligned} &\dot{\hat{x}} = \mathbf{A}\hat{x} + \mathbf{b}u + \mathbf{L}_x\nu_1, \\ &\nu_1(s) = V(s)(y - \mathbf{C}\hat{x}),\ r = e_y = (y - C\hat{x}), \end{aligned} \quad (13)$$

where scalar r is the residual signal, and the rest variables are described above. The residual signal must be sensitive to the constsnt fault f and senseless with respect to periodic disturbance d. Note that the adaptive observer-filter (13) is the basis for a solution of the main problem. Let us describe an error dynamics of the observation (1) by the model (13) in frequency domain of the form

$$A(s)e_y(s) = -L(s)\nu_1 + (P(s)d(s) + E(s)f(s)), \quad (14)$$

using the notations (3), (6). We can reduce (14) to the form

$$\begin{aligned} &A(s)e_y(s) = -\tilde{\nu}_1 + (P(s)d(s) + E(s)f(s)), \\ &\tilde{\nu}_1 = \tilde{V}(s)e_y(s) = \frac{\tilde{V}_1(s)}{V_2(s)}e_y(s) = L(s)V(s)e_y(s), \end{aligned} \quad (15)$$

using new corrective term $\tilde{\nu}_1(s)$. Let us consider transfer functions $F_{e_y d}(s)$, $F_{e_y f}(s)$ from d and f to e_y $F_{e_y d}(s) = P(s)V_2(s)/\Delta_1(s)$, $F_{e_y f}(s) = E(s)V_2(s)/\Delta_1(s)$, where $\Delta_1(s)$ is the characteristic polynomial of the closed-loop connection (15)

$$\Delta_1(s) = A(s)V_2(s) + \tilde{V}_1(s). \quad (16)$$

Transfer function $\tilde{V}(s) = \tilde{V}_1(s)/V_2(s)$ should be chosen to maximize the functional

$$J_1(\tilde{V}) = J_1^1(\tilde{V})/J_1^2(\tilde{V}), \text{ where} \quad (17)$$

$$J_1^1(\tilde{V}) = \left|F_{e_y f}(0)\right|,\ J_1^2(\tilde{V}) = \left|F_{e_y d}(\beta_0)\right|.$$

The given statement of the proposed problem is close to well-known H_-/H_∞ optimization approach [1]. Remark that this problem of simultaneous search of the control action vector and the optimal transfer function has an analytical solution.

One can easy see that the frequency response $\left|F_{e_y d}(j\omega)\right|$ has a dip in the neighborhood of β_0 to minimize $J_1^2(\tilde{V})$, i.e. $V_2(j\beta_0) \approx 0$. As a result, obtain

$$V_2(s) = \tilde{V}_2(s)\bar{V}_2(s,\beta_0),\ \bar{V}_2(s,\beta_0) = (s^2 + 2\varepsilon s + \beta_0^2), \quad (18)$$

where ε is a positive constant value close to zero. Let us consider the problem of $J_1^1(\tilde{V})$ maximization. The theorem of the root distribution [6] states that any polynomial $\Delta(s)$, $deg\,\Delta(s) = m_d$ with the degree of stability $\alpha_{st} > 0$ has the corresponding vector $\gamma \in R^{m_d}$, such as $\Delta(s) \equiv \Delta^*(s,\gamma)$, where

$$\Delta^*(s,\gamma) = \begin{cases} \tilde{\Delta}^*(s,\gamma), \\ (s + a_{d+1}(\gamma,\alpha_{st}))\tilde{\Delta}^*(s,\gamma), \end{cases} \quad (19)$$

$$\begin{aligned} &\tilde{\Delta}^*(s,\gamma) = \prod_{i=1}^{d}(s^2 + a_i^1(\gamma,\alpha_{st})s + a_i^0(\gamma,\alpha_{st})), \\ &a_i^1(\gamma,\alpha_{st}) = 2\alpha_{st} + \gamma_{i1}^2,\ a_i^0(\gamma,\alpha_{st}) = \alpha_{st}^2 + \gamma_{i1}^2\alpha_{st} + \gamma_{i2}^2, \\ &a_{d+1}(\gamma,\alpha_{st}) = \gamma_{d0}^2 + \alpha_{st},\ d = [m_d/2], \\ &\gamma = \{\gamma_{11},\gamma_{12},\gamma_{21},\gamma_{22},\ldots,\gamma_{d1},\gamma_{d2},\gamma_{d0}\}. \end{aligned} \quad (20)$$

There exists the relationship between the degree of stability and sensitivity to constant disturbance action (which should be maximized). Note that the value $\left|F_{e_y f}(0)\right|$ is inversely proportional to $\Delta_1(0)$ and let define $\deg\Delta_1(s) = n_1$ and its degree of stability as α_{st}. In accordance to the mentioned theorem, there exists γ^*, such as $\Delta_1(s) = \Delta_1^*(s,\gamma)$, constructed with formulas (19), (20), and

$$\Delta_1^*(0,\gamma^*) = \prod_{i=1}^{n_1} a_i^0(\gamma^*,\alpha_{st}) = \prod_{i=1}^{n_1}(\alpha_{st}^2 + \gamma_{i1}^{*2}\alpha_{st} + \gamma_{i2}^{*2}) \ge \alpha_{st}^{2n_1}.$$

This means that the value $|F_{e_y f}(0)|$ is inversely proportional to α_{st}. We use modal parametric synthesis approach, allowing to set the degree of stability for the polynomial (16) initially. Degree of the polynomial $L(s)$ is taken equal to $n-1$. Let formulate the algorithm of the observer (13) design.

Algorithm 1:

1. Define degrees of the polynomials $\tilde{V}_1(s)$, $V_2(s)$ in (18) and therefore $n_1 = deg\Delta_1(s)$, set α_{st}, $\gamma_1^0 \in R^{n_1}$, compute $\Delta_1^*(s,\gamma_1^0)$ by formulas (19), (20).

2. Solve the polynomial equation

$$A(s)V_2(s) + \tilde{V}_1(s) = \Delta_1^*(s,\gamma_1^0).$$

3. Choose any $n-1$ roots ξ_i, $i=\overline{1,n-1}$ of the polynomial $\tilde{V}_1(s)$, then calculate $L(s) = \prod_{i=1}^{n-1}(s-\xi_i)$, corresponding $\mathbf{L}_x$, such that $\mathbf{C}(\mathbf{I}s-\mathbf{A})^{-1}\mathbf{L}_x = L(s)$, and $V_1(s) = \tilde{V}_1(s)/L(s)$. Then evaluate $J_1(\tilde{V})$ by (17).

4. Maximize $J_1(\tilde{V})$ (17), repeating steps 2-4, using any numerical method. Use optimal $\gamma_1 = \gamma_1^*$ to compute required items $V_0(s) = V_1^0(s)/V_2^0(s)$ and $\mathbf{L}_{x0}$.

4. FAULT ESTIMATION

The object of this section is to construct the fault estimation observer (2), using the vector $\mathbf{L}_{x0}$, and the transfer matrix $V_0(s) = V_1^0(s)/V_2^0(s)$, computed above. Now let rewrite the expression (5) of the model (4):

$$A_1(s)e_y(s) = E_1(s)\nu_2(s) + P_1(s)d(s), \text{ where}$$

$$A_1(s) = sA(s)V_2^0(s) + sL_0(s)V_1^0(s), L_0(s) = \mathbf{C}(\mathbf{I}s-\mathbf{A})^{-1}\mathbf{L}_{x0},$$
$$P_1(s) = sP(s)V_2^0(s)(s), E_1(s) = -E(s)V_2^0(s).$$

Note that spectral density of d is presented by (9). The problem (7) of sensitivity minimization for the disturbance d with the central frequency β_0 can be formulated as the following mean-square optimization problem:

$$\tilde{J} = \tilde{J}(W) = \lim_{T\to\infty}\frac{1}{T}\int_0^T (e_y^2 + k^2\nu_2^2)dt \to \min_W,$$

where the parameter k characterizes a balance between the sensitivity and stability degree of the system. If we present d in the form (9), the special approach to H_2 optimization, described in papers [4, 5], can be used to solve this problem. Nevertheless, there is one serious trouble: the polynomial $N_2(s)$, which is determined by the formula

$$N_2(s)N_2(-s) = -s^2P(s)P(-s)V_2^0(s)V_2^0(-s)\times \times N_d(s)N_d(-s), \quad (21)$$

used in the following calculations, is submultiple for the characteristic polynomial (8) of the system closed by the optimal $W(s)$, computed in accordance with [4]. One can easy see that to provide the asymptotical stability, the factor s in (21) must be changed to $(s+p)$, where $p>0$ is a small number, i.e. N_2 must be changed to $\tilde{N}_2(s)$ as follows:

$$\tilde{N}_2(s)\tilde{N}_2(-s) = (s+p)(-s+p)P(s)P(-s)\times \times V_2(s)V_2(-s)N_d(s)N_d(-s). \quad (22)$$

Let us also note, that the parameter α in the (9) should be set close to zero to save frequency properties.

It is necessary to pay attention to the fact, mentioned in paper [5], that computed transfer function may be improper that obstructs its practical realization. One of ways to avoid this difficulty, is to deform the spectral power density (22) $\tilde{N}_2^*(s) = \tilde{N}_2(s)N^*(s)$, where $N^*(s)$ is a Hurwitz polynomial of the degree n^* by the way pointed in [5]. To save degree of stability of the closed-loop system, we accept it equal to p for $N^*(s)$ and parameterize $N^*(s)$ as $N^*(s) = N^*(s,\gamma_2)$, where $\gamma_2 \in R^{n^*}$, using (19), (20). As a result, we have

$$\tilde{N}_2^*(s) = \tilde{N}_2^*(s,\gamma_2) = \tilde{N}_2N^*(s,\gamma_2).$$

Process of $W(s)$ computation consists of the following steps:

1. Execute the factorization of the polynomial:

$$k^2A_1(s)A_1(-s) + E_1(s)E_1(-s) \equiv G(s)G(-s), \quad (23)$$

2. Construct the auxiliary polynomial

$$R(s) = \sum_{i=1}^{n}\frac{G(-s)}{g_i - s}\frac{E_1(-g_i)\tilde{N}_2^*(g_i)}{A_1(g_i)T(g_i)G'(-g_i)}, \quad (24)$$

where g_i, $i=\overline{1,n}$ are the distinct roots of $G(-s)$.

3. Represent a transfer function of the optimal filter

$$W(s) = \frac{[A_1(s)T(s)R(s) + E_1(-s)\tilde{N}_2^*(s)]/G(-s)}{[E_1(s)T(s)R(s) - k^2A_1(-s)\tilde{N}_2^*(s)]/G(-s)}, \quad (25)$$

where division to polynomial $G(-s)$ is done totally.

The optimal $W(s)$ is a function of the parameters γ_2, k, p. As a result, solution of the problem (12) is equal to minimization of the following functional

$$J_2(W) = J_2(\gamma_2,k,p) = T_p - T_p^0 + |T_p - T_p^0| + \ldots \ldots + J_p - J_p^0 + |J_p - J_p^0| + J_\omega - J_\omega^0 + |J_\omega - J_\omega^0|, \quad (26)$$

where $J_\omega = J_\omega(\gamma_2,k,p)$ is described by (11), $T_p = T_p(\gamma_2,k,p)$ is the time of fault estimation, and $J_p = J_p(\gamma_2,k,p)$ is the overshoot. Optimal parameters $k = k^*$, $p = p^*$ and $\gamma_2 = \gamma_2^*$ can be designed with the following algorithm.

Algorithm 2.

1. Set initial $k = k_0$, $p = p_0$ and vector $\gamma_2 = \gamma_2^0 \in R^{n^*}$. Compute the polynomial $\tilde{N}_2^*(s) = \tilde{N}_2^*(s, \gamma_2)$.

2. Compute the function $W(s)$, using formulas (23-25).

3. Evaluate $J_2(\gamma_2, k, p)$ (26).

4. Obtain vector $\gamma_2 = \gamma_2^*$, minimizing $J_2(\gamma_2^*, k, p)$ (26), using any numerical method, e.g. Nelder-Mead algorithm. If $J_2(\gamma_2^*, k, p)$ is not close to zero, then repeat steps 2-4 with new parameters k, p, searched, e.g., with enumeration.

5. Compute optimal $W_0(s) = W_1^0(s) / W_2^0(s)$.

5. NUMERICAL EXAMPLE

Consider the model (1) of the yaw ship motion with the constant speed, consisting of the following matrices

$$\mathbf{A} = \begin{pmatrix} -0.0936 & 0.634 & 0 \\ 0.048 & -0.0717 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \ \mathbf{b} = \begin{pmatrix} 0.0196 \\ 0.0160 \\ 0 \end{pmatrix}, \ \mathbf{E} = \mathbf{b},$$

$$\mathbf{P} = (0.41 \quad 0.0076 \quad 0)^T, \ \mathbf{c} = (0 \quad 0 \quad 1),$$

and the external disturbance (10) with $\beta_0 = 0.45$:

$$d(t) = 0.1sin(0.9\beta_0 t) + 1sin(\beta_0 t) + 0.1sin(1.1\beta_0 t).$$

Firstly, we compute parameters of the fault detection observer (13), where $V_1(s)$, $V_2(s)$ are polynomials of second degree $V_2(s) = \bar{V}_2(s) = s^2 + 0.002s + 0.2025, \tilde{V}_2(s) = 1$. Setting $\alpha_{st} = 0.03$, $\gamma_1^0 = (11111)$ we use the Algorithm 1 and obtain

$$V_0(s) = \frac{V_1^0(s)}{V_2^0(s)} = \frac{4.46 + 1.99s + 0.28}{s^2 + 0.002s + 0.2025},$$

$$\mathbf{L_{x0}} = (-0.87 \quad 0.46 \quad 1)^T.$$

The following parameters of the fault estimation observer are computed in accordance with the Algorithm 2. We accept $n^* = 3$ and use the initial parameter vector $\gamma_2^0 = (111)$. As a result, the optimal terms are the following:

$$k^* = 0.1, p^* = 0.02, \ N^*(s) = s^3 + 1.79s^2 + 2.15s + 1.45,$$

$$W_1^0(s) = 99.09\,s^7 + 542\,s^6 + 843.9\,s^5 + 449.3\,s^4 +$$

$$+ 212.2\,s^3 + 68.49\,s^2 + 8.788\,s + 0.158,$$

$$W_2^0(s) = s^7 + 4.646\,s^6 + 7.948\,s^5 + 7.858\,s^4 + 6.512\,s^3$$

$$+ 1.863\,s^2 + 0.9995\,s + 0.09226.$$

Figures 1, 2 illustrate frequency response $A(\omega)$ of the $F_{e_f d}(s)$ (7) for the designed observer and the fault estimation process. The dip in the area of the central frequency β_0 can be seen in the Fig. 1. $A(\omega)$ can be compared with the response $A_0(\omega)$ of the observer (3) with the constant parameters $\mathbf{L_x} = (36.0 \quad 23.0 \quad 6.8)^T$, $V(s) \equiv 1$, $W(s) \equiv 3162.3$ and structure described in [2,3].

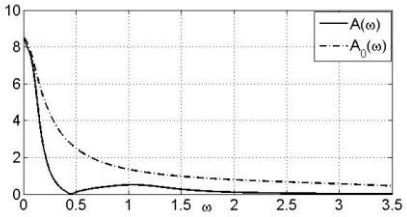


Figure 1. Frequency responses $\mathbf{A(\omega)}$ and $\mathbf{A_0(\omega)}$.

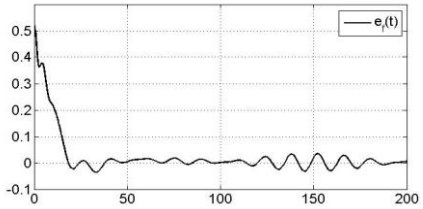


Figure 2. Fault estimation process (error dynamics).

6. CONCLUSION

In this paper, a novel approach to the fault estimation has been presented. The simulated result demonstrates its applicability with respect to the external disturbance with the given frequency range. Sincerely, proposed approach has a serious disadvantage: it cannot be applied to rapidly varying fault estimation. The object of the future research is spectral solution of such faults estimation, maybe taking into account time delays or robust features.

7. REFERENCES

[1] Ding S. 2008. Model-based fault diagnosis techniques: design schemes, algorithms, and tools. Springer Science & Business Media.

[2] Zhang, K., Jiang, B., & Shi, P. (2009). Fast fault estimation and accommodation for dynamical systems. *IET Control Theory & Applications*, 3(2), 189-199. DOI=http://dx.doi.org/10.1049/iet-cta:20070283.

[3] Zhang, K., Jiang, B. and Shi, P. 2013. Observer-Based Fault Estimation and Accommodation for Dynamic Systems. Springer-Verlag. Berlin. DOI=http://dx.doi.org/10.1007/978-3-642-33986-8.

[4] Veremey, E.I. 2015. Efficient Spectral Approach to SISO Problems of H2-Optimal Synthesis. In *Applied Mathematical Sciences*, 9(79), 3897-3909. http://dx.doi.org/10.12988/ams.2015.54335.

[5] Veremey, E.I. and Knyazkin Y.V. 2016. Spectral H_2 optimal correction of additive fault estimation observer. In *ITM Web of Conferences* (Vol. 6). EDP Sciences. DOI=http://dx.doi.org/10.1051/itmconf/20160601005.

[6] Veremey, E.I., Smirnov, M.N. and Smirnova, M.A. 2015. Synthesis of stabilizing control laws with uncertain disturbances for marine vessels. In *Proceedings of the Stability and Control Processes" in Memory of VI Zubov (SCP), 2015 International Conference (pp. 1-3).* DOI=http://dx.doi.org/10.1109/SCP.2015.7342219.

Map-Reduce Based Algorithm for the Analysis of Vital Variables of Neonatal Patients

Sipan Arevshatyan
Department of Computing Engineering, Universitat Politècnica de València
sipan443@gmail.com

Houcine Hassan
Department of Computing Engineering, Universitat Politècnica de València
husein@disca.upv.es

Carlos Domínguez
Department of Computing Engineering, Universitat Politècnica de València
carlosd@disca.upv.es

ABSTRACT

This paper describes neonatal healthcare problems and shows ways to analyse the information using modern technologies like Big Data, Hadoop, and Map Reduce programming in order to help doctors to solve these problems. In the work we have evaluated 4 characteristics of 10 patients by designing an algorithm to analyse the data. The paper consists of four parts: Introduction, related work, application problem and evaluation and conclusion. In these chapters, we explain the concept, characteristics and need of Big Data, state the problem in neonatal health care, provide useful information about related works, show that Big Data do exist in hospitals and conclude the work.

Keywords

Neonatal health care, Big Data, Data analysis, Hadoop, Map Reduce programming

1. INTRODUCTION

Due to the human activities, a lot of information is being created every day. One of the most interesting field of these activities can be considers health care and particularly neonatal health care as there are a lot of neonates who are born with some health problems.

Thus, doctors need to use the information coming from NICU tools in order to make right decisions. Sometimes the information is so big and it is useful to use modern technologies in order to achieve good results. In this point of view, we can say that doctors usually are dealing with Big Data. Big Data is a term for massive data sets having large and complex structure with the difficulties of capturing, storing, searching, sharing, analysing, transferring and visualizing for the future processes [11]. Big data in neonatal health care can be described by three main components: variety, velocity and volume.

*ICAIT'*16, Oct. 6–8, 2016, Aizu-Wakamatsu, Japan.

Volume: The quantity of generated data in NICU is important in this context. The size of the data determines the value and potential of the data under consideration, and whether it can actually be considered Big Data or not. The name 'Big Data' itself contains a term related to size, and hence the characteristic.

Velocity: In this context, the speed at which the data is generated and processed to meet the demands and the challenges that lie in the path of growth and development. For example, smart infusion pumps (SIPs) contribute can provide more than 60 different types of data every 10 seconds.

Variety: The type and nature of the data. This helps people who are associated with and analyse the data to effectively use the data to their advantage and thus uphold its importance.

2. RELATED WORK

The market for healthcare services has increased exponentially. This is due to the growing tendency for personal healthcare to move away from the traditional hubs of healthcare, such as hospitals and clinics, to the private home and especially the mobile environment. In most developed countries an aging population contributes to the growth in the demand for distributed healthcare services. As a result of the nature of healthcare, the precision and real-time delivery of data is crucial. To fulfil all these requirements, advanced and smart technologies should be applied.

One of the very smart technologies was designed by Carolyn McGregor, University of Ontario Institute of Technology, Canada. The platform is called Artemis. Artemis is an online health analytics platform that enables concurrent diagnoses of multiple patients through real-time analysis of multiple data streams [1][7]. It supports acquisition and storage of patients´ information for the purpose of online analytics. Artemis is currently implemented in and used to help sick children in Ontario, Canada and the research team is going to deploy the platform in other cities of Canada as well as in China and Australia.

Another research made in USA by Rollins School of Public Health, Emory University, USA shows the importance of maternal health during pregnancy [2]. The research was held in 13 states of the USA. It states that smoking increased infant length of stay by 1.1%. NICU infants cost $2496 per night while in the NICU and $1796 while in a regular nursery compared to only $748 for non-NICU infants. Multivariate analysis is used to estimate the relationship of smoking to probability of admission to an NICU and, separately, the length of stay for those admitted or not admitted to an NICU.

3. APPLICATION PROBLEM AND EVALUATION

3.1 Health Care and Big Data

Probably some years ago, one might not expect these two areas might to even be mentioned in the same sentence. But now they are coming together and tend to change the face of the medicine. The coming together of healthcare and Big Data means higher tech medical solutions for the general problems in medicine. Particularly, high tech medical solutions could be implemented in neonatal health care and medicine [5].

In this article the problem is related to the analysis of vital variables of neonates in hospital.

A neonatal intensive-care unit (NICU), also known as an intensive care nursery (ICN), is an intensive-care unit specializing in the care of ill or premature new born infants. While a child is inside of NICU medical devices generate a lot of information. Devices monitor physiological data streams that reflect the functioning of vital organs while others provide ventilation support. Sometimes the information is so big that we can say that doctors deal with Big Data. In this point of view it is important to gather the information and use it in order to make decisions:

The information coming from NICU can vary. Here are some examples of the Bid Data in NICU: Many NICU patients have heart activity monitored by electrocardiography (ECG), which can sample up to 1,000 readings a second to construct a waveform signal demonstrating the functioning of the heart. This translates to 86.4 million readings a day per patient. From this source signal, the ECG device also derives the heart rate and respiration rate, with each of these signals producing 86,400 readings a day per patient [4][8].

Drug and nutrition infusion data from smart infusion pumps (SIPs) contribute to the big data problem. SIPs can provide more than 60 different types of data every 10 seconds. If a new born stays in the NICU for 30 days, one SIP can generate 4.4 Mbytes of data per hour, 106 Mbytes of data per day, and 3 Gbytes of data monthly. Preterm infants can be connected to up to 13 SIPs, resulting in 39 Gbytes of drug infusion data from a single patient per month. And a lot of information could be generated and this information should be processed in order to help and survive new born children [6].

It's very important to mention that maternal health is closely linked to new born survival. While great strides have been made in reducing global child mortality, new-borns now account for 44 percent of all childhood deaths. Each year, 2.9 million new-borns needlessly die within their first month and an additional 2.6 million are stillborn. The main causes, which are preventable and treatable, are complications due to prematurity, complications during delivery, and infection [3].

The analysis showed that maternal smoking increased the relative risk of admission to an NICU by almost 20%. For infants admitted to the NICU, maternal smoking increased length of stay while for non-NICU infants it appeared to lower it. Over all births, however, smoking increased infant length of stay by 1.1%.

3.2 Healthcare Application

For this research, we will use 4 main characteristics to analyse: heart rate, respiratory rate, lower blood pressure and upper blood pressure. The data are generated randomly. We suppose that this data are generated by NICU. Diagram 1, diagram 2, diagram 3 and diagram 4 show how heart rate, respiration rate, lower blood pressure and upper blood pressure are being changed during the time.

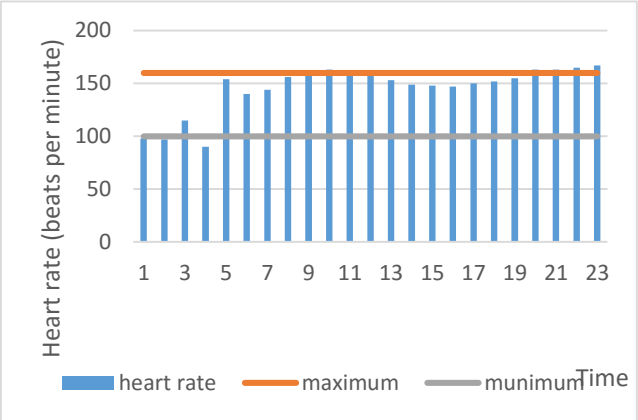


Diagram 1: Heart Rate

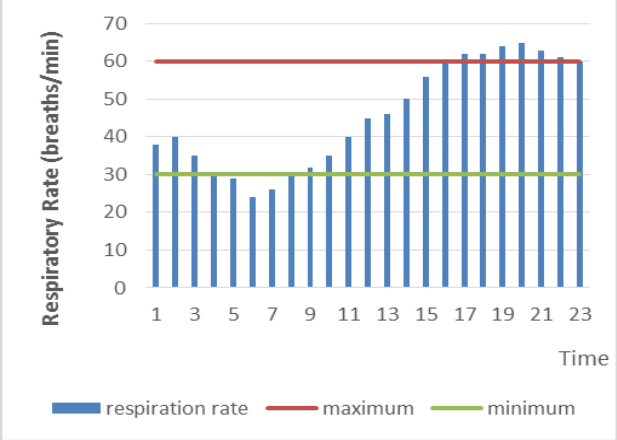


Diagram 2: Respiratory Rate

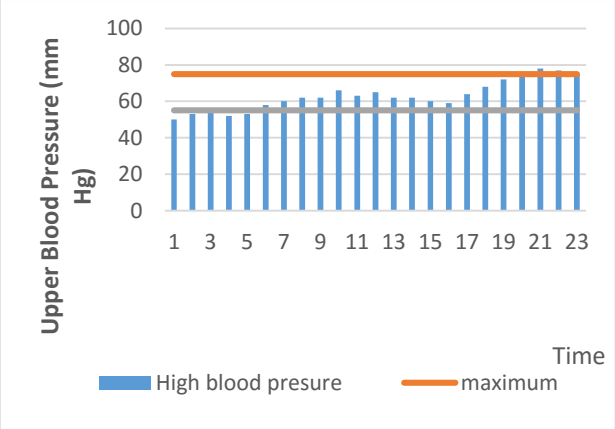


Diagram 3: Upper Blood Pressure (mm Hg)

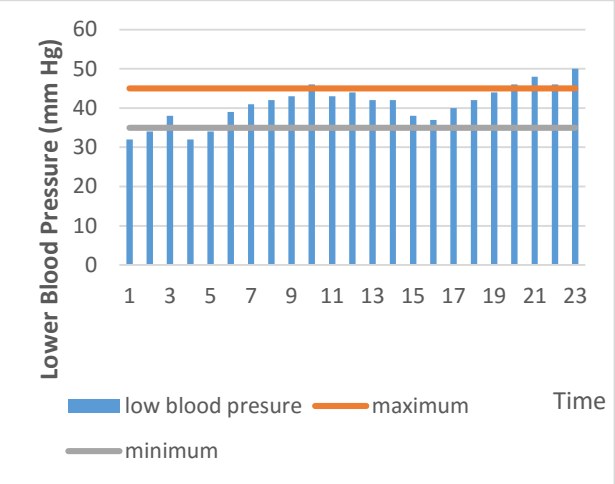


Diagram 4: Lower Blood Pressure (mm Hg)

Then we use Hadoop and Map Reduce programming in order to extract information for the patient or patients. A large part of the power of Map Reduce comes from its simplicity. Map Reduce works by breaking the processing into two phases: the map phase and the reduce phase [9][10][12]. Each phase has key-value pairs as input and output, the types of which may be chosen by the programmer. The programmer also specifies two functions: the map function and the reduce function. Below is presented the algorithm of simple program Word Count as well as one example of the demonstration of the program, which can be implied into the hospital.

3.3 Proposed Algorithm

The mapper emits an intermediate key-value pair for each word in a document.

The reducer sums-up all the counts for each word.

1: **class** Mapper

2: **method** Map(docid a; doc d)

3: **for all** term t ∈ doc d do

4: **Emit**(term t; count)

1: **class** Reducer

2: **method** Reduce(term t; counts [c1; c2; : : :])

3: sum = 0

4: **for all** count c ∈ counts [c1; c2; : : :] do

5: sum = sum + c

6: Emit(term t; count sum)

To visualize the way the map works, consider the following sample lines of input data.

name0-96-43-27-22

name2-77-170-4-28

name1-75-80-103-21

name0-76-42-28-9

name2-74-188-77-30

name1-75-70-91-67

name0-69-173-80-36

name2-75-47-7-36

name2-87-101-23-32

name1-89-93-40-45

Etc...

Here the lines present records of the patients. Every part of the line has its own meaning: Thus, first part is the name; second part is the heart rate, then upper blood pressure and lower blood pressure and finally respiration rate.

These lines are presented to the map function as the key-value pairs:

(0, name0-96-43-27-22)

(16, name2-77-170-4-28)

(32, name1-75-80-103-21) Etc...

The keys are the line offsets within the file, which we ignore in our map function. The map function merely extracts the name and the heart rate (indicated in bold), and emits them as its output (the heart rate values have been interpreted as integers):

(name0, 96); (name2, 77); (name1, 75) etc...

The output from the map function is processed by the Map Reduce framework before being sent to the reduce function. This processing sorts and groups the key-value pairs by key. So, continuing the example, the reduce function sees the following input:

(name0, [96, 76, 69]); (name1, [75, 75, 89]); (name2, [77, 74, 75, 87])

Each name appears with a list of all its heart rate readings. All the reduce function has to do now is iterate through the list and pick up the maximum reading:

(name0, 96); (name1, 89); (name2, 87)

This is the final output: the maximum heart rate for each patient. The information can be saved as a txt file and will be ready for future use and analysis.

4. EXPERIMENTAL RESULTS

In this research, we used Intel® Core™ i5-660, 3.33GHz×4, Memory 3.7GB computer. In the file, which is going to be analysed, we have millions of lines. The size of the file is 37.9MB (1 million lines). This file contains 4 characteristics of 10 patients. After the running of the program, we can see that the map reduce program works correctly and emits data we need. In this case, we see the maximum heart rate for each patient. The same algorithm could be used to emit data about other characteristics (respiration rate, lower blood pressure etc.).

The doctor also can estimate the time. In this case, we see that the CPU time spent is 9480 ms. In order to decrease the time we only should add computers.

In addition to this we offer the equation below to evaluate the state of the patient:

$$F = \alpha X + \beta Y + \gamma Z + \theta Q$$

Where:

$$X = \frac{X_1}{X_{\max}}; \quad Y = \frac{Y_1}{Y_{\max}}; \quad Z = \frac{Z_1}{Z_{\max}}; \quad Q = \frac{Q_1}{Q_{\max}}$$

X_{max} – maximum heart rate of the patients

Y_{max} – maximum respiratory rate of the patients

Z_{max} – maximum upper blood pressure or the patients

Q_{max} – maximum lower blood pressure of the patients

X_1, Y_1, Z_1 and Q_1 are current heart rate, respiratory rate, upper blood pressure and lower blood pressure representatively.

In the formula the sum of the coefficients (α, β, γ, θ) is 1. It will make the F formula give us a value between 0 and 1. In this paper we consider that α=β=γ=θ but later the exact values might be defined more precisely by working with IT, natural science and medicine researchers. As we have 4 characteristics (n=4) the coefficients can be evaluated in this way:

$$\alpha=\beta=\gamma=\theta=1/n;$$

So we will get that α=β=γ=θ=0.25.

In order to evaluate patients' condition we consider that:

$$F \in [0, 0.3] \rightarrow Bad\ condition$$

$$F \in [0.31, 0.8] \rightarrow Normal\ condition$$

$$F \in [0.81, 1] \rightarrow Dangerous\ condition$$

For example let us assume that X_1= 98, Y_1=38, Z_1=50 and Q_1=32; according to the database X_{max}= 200, Y_{max}=80, Z_{max}=100, Q_{max}=50.

$$F = 0.25 * \frac{98}{200} + 0.25 * \frac{38}{80} + 0.25 * \frac{50}{100} + 0.25 * \frac{32}{50} \approx 0.52$$

As F≈0.52 it means that the condition of the patient is normal.

5. CONCLUSION

Recently we have seen huge advances in the amount of data we generate and collect, as well as our ability to use technology to analyse and understand it. One of the most interesting field Big Data can occur is health care. Big Data in healthcare is being used to predict epidemics, cure disease, improve quality of life and avoid preventable deaths. To reach the goals doctors in the hospital and developers should work together to facilitate the creation of the platform as well as for privacy reasons when the information of the patients is collected. In this paper we have presented an application to deal with Big Data in health care. We have defined 4 variables related to patients. We have proposed an algorithm to analyse the values of these variables. Finally, it is presented an analytical formula to estimate the health of the patients depending on the current values of the variables.

6. REFERENCES

[1] McGregor, C, Catley, C., James, A., & Padbury, J. (2011). Next Generation Neonatal Health Informatics with Artemis. Medical Informatics Europe (MIE) 2011.

[2] Adams EK1, Miller VP, Ernst C, Nishimura BK, Melvin C, Merritt R. 2002, Neonatal health care costs related to smoking during pregnancy

[3] http://www.gatesfoundation.org/What-We-Do/Global-Development/Maternal-Newborn-and-Child-Health

[4] Carolyn McGregor, University of Ontario Institute of Technology, Canada. Big Data in Neonatal Intensive Care

[5] Bernard Marr, April 2015, Forbs. How Big Data Is Changing Healthcare. http://www.forbes.com/sites/bernardmarr/2015/04/21/how-big-data-is-changing-healthcare/#6dcf69cf32d9

[6] C. McGregor et al., "Late Onset Neonatal Sepsis Detection in Newborn Infants via Multiple Physiological Streams," J. Critical Care, vol. 28, no. 1, 2013, pp. e11-e12.

[7] M. Blount et al., "Real-Time Analysis for Intensive Care: Development and Deployment of the Artemis Analytic System," IEEE Eng. in Medicine and Biology Magazine, vol. 2010, pg 110 – 118

[8] Carolyn McGregora, Andrew Jamesb,c,a, Mike Eklundd, Daby Sowe, Maria Eblinge, Marion Blounte © 2013 IMIA and IOS Press. Real-time Multidimensional Temporal Analysis of Complex High Volume Physiological Data Streams in the Neonatal Intensive Care Unit

[9] Hadoop: The Definitive Guide, Tom White, 2012

[10] Hadoop: Open source implementation of MapReduce. http://lucene.apache.org/

[11] http://bigdatauniversity.com/

[12] http://hadoop.apache.org

On the Multifacility Weber Problem for Four Terminals

Alexei Yu. Uteshev
St. Petersburg State University
7/9, Universitetskaya nab.
St. Petersburg, 199034, Russia
alexeiuteshev@gmail.com

Elizaveta A. Semenova
St. Petersburg State University
7/9, Universitetskaya nab.
St. Petersburg, 199034, Russia
semenova.elissaveta@gmail.com

ABSTRACT

We study a generalization of the Euclidean minimal tree problem to the case of the planar weighted networks consisting of four given terminals and two extra facilities. Explicit analytical formulae are presented for the conditions of the existence of the network, facility coordinates and for the total network cost. These formulae are utilized for the investigation of the network dynamics under variation of parameters.

Categories and Subject Descriptors

G.2.2 [**Discrete mathematics**]: Graph Theory—*Network problems*; G.1.6 [**Numerical analysis**]: Optimization—*Global optimization*; I.1.2 [**Symbolic and algebraic manipulation**]: Algorithms—*Algebraic Algorithms*

General Terms

Theory, Algorithms

Keywords

Euclidean multifacility location problem, Weber problem, nonlinear optimization, analytical solution

1. INTRODUCTION

The classical *Weber* or *generalized Fermat-Torricelli problem* is stated as that of finding the point (facility, junction) $S = (x_*, y_*)$ that minimizes the sum of weighted distances from itself to $n \geq 3$ fixed points (terminals) $\{P_j = (x_j, y_j)\}_{j=1}^n$ in the Euclidean plane:

$$\min_{S\in\mathbb{R}^2} \sum_{j=1}^{n} m_j |SP_j| \,. \tag{1}$$

Hereinafter $|\cdot|$ stands for the Euclidean distance and the weights $\{m_j\}_{j=1}^n$ are assumed to be positive real numbers.

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

This nonlinear optimization problem was stated by Alfred Weber [6] with regard to the optimal facility location problem like the one of finding the optimal, i.e., minimizing the transportation costs, position for the plant manufacturing one ton of the final product from $\{m_j\}_{j=1}^n$ tons of distinct raw materials located at $\{P_j\}_{j=1}^n$. He also treated the *multifacility problem* consisting in finding the set of $\ell \geq 2$ facility points $\{S_i\}_{i=1}^\ell$ in $\mathbb{R}^2$ connected to the terminals $\{P_j\}_{j=1}^n$ that solves the optimization problem

$$\min_{\{S_1,\ldots,S_\ell\}\subset\mathbb{R}^2} \left\{ \sum_{j=1}^{n}\sum_{i=1}^{\ell} m_{ij}|S_iP_j| + \sum_{k=1}^{\ell}\sum_{i=k+1}^{\ell-1} \widetilde{m}_{ik}|S_iS_k| \right\}.$$

This problem can be considered as a natural generalization of the celebrated *Steiner minimal tree problem* aimed at construction of the network of the minimal length linking the given terminals.

Dozens of papers are devoted to the Weber problem, its ramifications and applications; we refer to [2] and [3] for the reviews. They are focused onto the application of the variety of numerical procedures for the nonlinear optimization problem. The main difficulty consists in the fact that the objective (or cost) function of the Weber problem is nondifferentiable, and the extensions of the standard nonlinear programming versions of iterative procedures for finding its minimum, like the gradient descent ones, should be modified. One of these modifications is based on the Weiszfeld algorithm.

The present paper is devoted to an alternative approach for the problem, namely an analytical one. We are looking for the explicit expressions for the facility coordinates as functions of the problem parameters (terminal coordinates and weights). This approach has been originated in the recent papers [4] and [5] where the unifacility Weber problem for three terminals and Steiner minimal tree problem for four terminals have been solved *by radicals*. Within the framework of this approach, we will be focused here on solution of the Weber problem for the case of $n = 4$ terminals and $\ell = 2$ facilities, namely we are looking for the facilities $S_1 = (x_*, y_*)$ and $S_2 = (x_{**}, y_{**})$ minimizing the following objective function

$$\begin{aligned} F(x_*, y_*, x_{**}, y_{**}) = {} & m_1|S_1P_1| + m_2|S_1P_2| \\ & + m_3|S_2P_3| + m_4|S_2P_4| + m|S_1S_2| \,. \end{aligned} \tag{2}$$

We prove that this problem can also be solved by radicals with the main result of paper formulated in the next section.

2. ANALYTICS

We will treat the case where the terminals $\{P_j\}_{j=1}^4$, while counted counterclockwise, compose a convex quadrilateral $P_1P_2P_3P_4$.

THEOREM 1. *The necessary condition for the existence of solution to the Weber problem is that of positivity of the values*

$$k_{12} = (m+m_1+m_2)(m-m_1+m_2) \times (m+m_1-m_2)(-m+m_1+m_2),$$

$$k_{34} = (m+m_3+m_4)(m-m_3+m_4) \times (m+m_3-m_4)(-m+m_3+m_4).$$

Set

$$\begin{aligned}\tau_1 &= \sqrt{k_{12}}[\sqrt{k_{34}}(x_4-x_3) - (m^2+m_3^2-m_4^2)y_3 \\ &\quad - (m^2-m_3^2+m_4^2)y_4] + 2m^2\sqrt{k_{12}}y_2 + k_{12}(x_1-x_2) \\ &\quad + (m^2+m_1^2-m_2^2)[\sqrt{k_{34}}(y_3-y_4) + (m^2+m_1^2-m_2^2)x_1 \\ &+(m^2-m_1^2+m_2^2)x_2-(m^2+m_3^2-m_4^2)x_3-(m^2-m_3^2+m_4^2)x_4],\end{aligned}$$

$$\begin{aligned}\tau_2 &= -\sqrt{k_{12}}[\sqrt{k_{34}}(x_4-x_3) - (m^2+m_3^2-m_4^2)y_3 \\ &\quad - (m^2-m_3^2+m_4^2)y_4] - 2m^2\sqrt{k_{12}}y_1 - k_{12}(x_1-x_2) \\ &\quad + (m^2-m_1^2+m_2^2)[\sqrt{k_{34}}(y_3-y_4) + (m^2+m_1^2-m_2^2)x_1 \\ &+(m^2-m_1^2+m_2^2)x_2-(m^2+m_3^2-m_4^2)x_3-(m^2-m_3^2+m_4^2)x_4],\end{aligned}$$

$$\begin{aligned}\eta_1 &= \sqrt{k_{12}}[\sqrt{k_{34}}(y_4-y_3) + (m^2+m_3^2-m_4^2)x_3 \\ &\quad + (m^2-m_3^2+m_4^2)x_4] - 2m^2\sqrt{k_{12}}x_2 + k_{12}(y_1-y_2) \\ &\quad + (m^2+m_1^2-m_2^2)[\sqrt{k_{34}}(x_4-x_3) + (m^2+m_1^2-m_2^2)y_1 \\ &+(m^2-m_1^2+m_2^2)y_2-(m^2+m_3^2-m_4^2)y_3-(m^2-m_3^2+m_4^2)y_4],\end{aligned}$$

$$\begin{aligned}\eta_2 &= -\sqrt{k_{12}}[\sqrt{k_{34}}(y_4-y_3) + (m^2+m_3^2-m_4^2)x_3 \\ &\quad + (m^2-m_3^2+m_4^2)x_4] + 2m^2\sqrt{k_{12}}x_1 - k_{12}(y_1-y_2) \\ &\quad + (m^2-m_1^2+m_2^2)[\sqrt{k_{34}}(x_4-x_3) + (m^2+m_1^2-m_2^2)y_1 \\ &+(m^2-m_1^2+m_2^2)y_2-(m^2+m_3^2-m_4^2)y_3-(m^2-m_3^2+m_4^2)y_4],\end{aligned}$$

and set the values for $\tau_3, \tau_4, \eta_3, \eta_4$ via the formulae obtained by cyclic substitution for subscripts

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 \end{pmatrix}$$

in the above expressions for $\tau_1, \tau_2, \eta_1, \eta_2$ correspondingly.

If all the values

$$\delta_1 = \eta_2(x_1-x_2) + \tau_2(y_2-y_1),$$

$$\delta_2 = \eta_1(x_1-x_2) + \tau_1(y_2-y_1),$$

$$\delta_3 = \eta_4(x_3-x_4) + \tau_4(y_4-y_3),$$

$$\delta_4 = \eta_3(x_3-x_4) + \tau_3(y_4-y_3),$$

$$\begin{aligned}\delta = &-\frac{\delta_1\left(m^2+m_1^2-m_2^2\right)}{\sqrt{k_{12}}} - \frac{\delta_3\left(m^2+m_3^2-m_4^2\right)}{\sqrt{k_{34}}} \\ &+(\eta_1+\eta_2)(y_1-y_3) + (\tau_1+\tau_2)(x_1-x_3)\end{aligned}$$

are positive then there exists a pair of points S_1 and S_2 lying inside $P_1P_2P_3P_4$ that furnishes the minimal value for (2).

The coordinates of point S_1 are as follows:

$$x_* = x_1 - \frac{2\delta_1 m^2\tau_1}{\sqrt{k_{34}}\left[(\eta_1+\eta_2)^2+(\tau_1+\tau_2)^2\right]}, \quad (3)$$

$$y_* = y_1 - \frac{2\delta_1 m^2\eta_1}{\sqrt{k_{34}}\left[(\eta_1+\eta_2)^2+(\tau_1+\tau_2)^2\right]} \quad (4)$$

while those of point S_2:

$$x_{**} = x_3 - \frac{2\delta_3 m^2\tau_3}{\sqrt{k_{12}}\left[(\eta_1+\eta_2)^2+(\tau_1+\tau_2)^2\right]}, \quad (5)$$

$$y_{**} = y_3 - \frac{2\delta_3 m^2\eta_3}{\sqrt{k_{12}}\left[(\eta_1+\eta_2)^2+(\tau_1+\tau_2)^2\right]}. \quad (6)$$

The minimal value for (2) *(cost of the network) then equals*

$$\mathfrak{C} = \frac{\sqrt{(\eta_1+\eta_2)^2+(\tau_1+\tau_2)^2}}{4m^3}. \quad (7)$$

Theorem 1 claims that, for the case of two facilities, the Weber problem can be solved by radicals. The proof is similar to its counterpart for the equal weighted case [5]. It can be proved that the {4 terminals, 2 facilities}-Weber problem can be reduced to a twain of {3 terminals, 1 facility}-Weber problems. For instance, the configuration of weights

$$\{P_1, m_1\}, \{P_2, m_2\}, \{Q, m\}$$

with

$$Q = 1/(2m^2)$$

$$\begin{aligned}\times \quad &\left(m^2(x_3+x_4) + (m_3^2-m_4^2)(x_3-x_4) - \sqrt{k_{34}}(y_3-y_4),\right. \\ &\left. m^2(y_3+y_4) + (m_3^2-m_4^2)(y_3-y_4) + \sqrt{k_{34}}(x_3-x_4)\right)\end{aligned}$$

possesses a solution to the unifacility problem coinciding with the position of the facility S_1. For this type of problems, an analytical solution is already constructed [4].

COROLLARY 1. *Under the conditions of Theorem 1, the point $S_1 = (x_*, y_*)$ lies inside the triangle $P_1P_2S_2$ and provides a solution to the unifacility Weber problem*

$$\min_{S\in\mathbb{R}^2}\left(m_1|SP_1| + m_2|SP_2| + m|SS_2|\right).$$

Similar statement is also valid for the point $S_2 = (x_{**}, y_{**})$ and the terminals P_3, P_4 and S_1.

We outline briefly the meaning of the conditions from Theorem 1. First, due to Heron's formula, the values $\frac{1}{4}\sqrt{k_{12}}$ and $\frac{1}{4}\sqrt{k_{34}}$ equal the squares of the so-called weight triangles, i. e. the triangles composed with the sets of edges coinciding with $\{m_1, m_2, m\}$ and $\{m_3, m_4, m\}$ respectively. The positivity of k_{12} and k_{34} guarantees the existence of both weight triangles. The condition for the positivity of all the delta values from the statement of Theorem 1 is essential for the problem solubility. Conditions $\{\delta_j > 0\}_{j=1}^4$ ensure the location of the facilities S_1 and S_2 inside the quadrilateral $P_1P_2P_3P_4$, while the condition $\delta > 0$ guarantees the facilities against their collision since

$$|S_1S_2| = \frac{\delta}{\sqrt{(\eta_1+\eta_2)^2+(\tau_1+\tau_2)^2}}. \quad (8)$$

COROLLARY 2. *For the equal weighted case* $\{m_j = 1\}_{j=1}^4$, $m = 1$, *the expression for* δ *can be represented in the form*

$$\delta = \frac{8}{\sqrt{3}}\,[x_3 - x_1, y_3 - y_1] \cdot \begin{bmatrix} \sqrt{3}/2 & 1/2 \\ -1/2 & \sqrt{3}/2 \end{bmatrix} \cdot \begin{bmatrix} x_4 - x_2 \\ y_4 - y_2 \end{bmatrix}.$$

This value is positive iff the angle between the diagonal $\overrightarrow{P_1P_3}$ *of the quadrilateral and the other diagonal* $\overrightarrow{P_2P_4}$ *turned through by* $\pi/6$ *clockwise is acute. Equivalently, if we denote by* ψ *the angle between diagonal vectors* $\overrightarrow{P_1P_3}$ *and* $\overrightarrow{P_2P_4}$ *then* δ *is positive iff* $\psi < \pi/2 + \pi/6 = 2\,\pi/3$. *This confirms the known condition for the existence of a full Steiner tree of the following topology:*

$$\begin{matrix} P_1 & & P_4 \\ & S_1S_2 & \\ P_2 & & P_3 \end{matrix}\ .$$

Formulae (3)-(6) yield then the coordinates of Steiner points while the length of the corresponding Steiner tree equals

$$\mathfrak{C} = \frac{1}{2}\sqrt{A^2 + B^2}$$

where

$$\begin{aligned} A &= \sqrt{3}(x_1 - x_2 - x_3 + x_4) + (y_1 + y_2 - y_3 - y_4), \\ B &= (x_1 + x_2 - x_3 - x_4) + \sqrt{3}(-y_1 + y_2 + y_3 - y_4)\,. \end{aligned}$$

3. EXAMPLES

In comparison with numerical (iterative) procedures for solving the Weber problem, representation of its solution in analytical form given in the previous section looks cumbersome. However, we give the following reasons for its utility:

- Though the numerical procedures are generically faster if dealing with particular specialization of the problem parameters, their approximation properties might be invalid when at least one of facilities being searched happens to lie close to a terminal position. On the contrary, analytical formulae are universal in the sense that they yield the exact result (i.e., free of truncation errors) regardless on the position of facilities.
- In case of the problem dealing with some variable parameters, analytics provide one with a unique opportunity to evaluate their influence on its solution. In particular, this means that the *bifurcation values* for the parameters can be determined responsible for the degeneracy of the network topology.

In the present section we will exemplify the latter point.

EXAMPLE 1. *Find the coordinates of facilities* S_1, S_2 *for the following configuration of weights*

$$\begin{matrix} P_1 = (1,5) \\ m_1 = 3 \end{matrix}\ \Big|\ \begin{matrix} P_2 = (2,1) \\ m_2 = 2 \end{matrix}\ \Big|\ \begin{matrix} P_3 = (7,2) \\ m_3 = 3 \end{matrix}\ \Big|\ \begin{matrix} P_4 = (6,7) \\ m_4 = 4 \end{matrix}\ \Big|\ m = 4\,.$$

Solution. The conditions of Theorem 1 are fulfilled: the values $\delta_1 \approx 14124, \delta_2 \approx 29388, \delta_3 \approx 34784, \delta_4 \approx 18831$ and $\delta \approx 11721$ are positive. Formulae (3)-(7) then give the coordinates for facilities

$$x_* = \frac{2266800 + 772027\sqrt{15} + 453552\sqrt{33} + 246177\sqrt{55}}{48\left(22049 + 2085\sqrt{15} + 945\sqrt{33} + 2559\sqrt{55}\right)} \approx 3.701271,$$

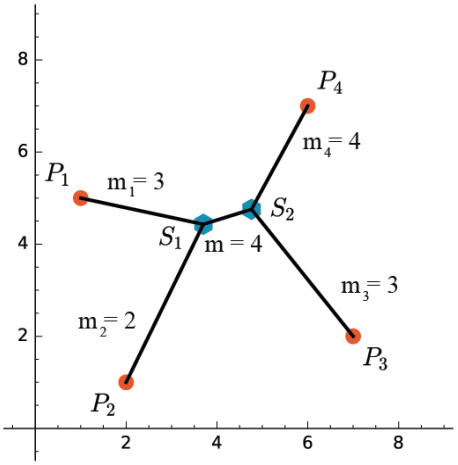


Figure 1: Network for the configuration of weights from Example 1

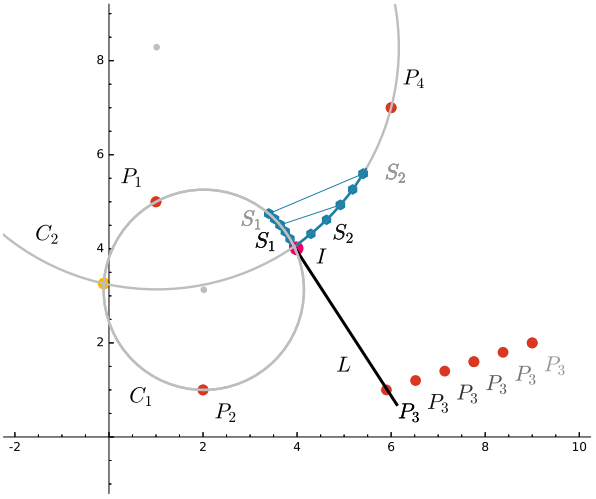


Figure 2: Dynamics of points S_1, S_2 under variation of terminal P_3.

$$y_* = \frac{1379951 + 201984\sqrt{15} + 97279\sqrt{33} + 154368\sqrt{55}}{16\left(22049 + 2085\sqrt{15} + 945\sqrt{33} + 2559\sqrt{55}\right)} \approx 4.430843,$$

$$x_{**} \approx 4.761622, \quad y_{**} \approx 4.756175,$$

and the cost of the network (Fig. 1):

$$\mathfrak{C} = \frac{1}{8}\sqrt{44098 + 4170\sqrt{15} + 5118\sqrt{55} + 1890\sqrt{33}} \approx 41.280608.$$

EXAMPLE 2. *For the terminals* P_1, P_2, P_4 *from Example 1 and for* P_3 *moving towards* P_2 *from the starting position at* $(9,2)$ *find the loci of facilities* S_1, S_2.

Solution. It turns out that when P_3 wanders, the facility S_j moves along the arc of the circle

$$C_j = \left\{(x,y) \in \mathbb{R}^2 \,\middle|\, (x - X_j)^2 + (y - Y_j)^2 = r_j^2\right\}.$$

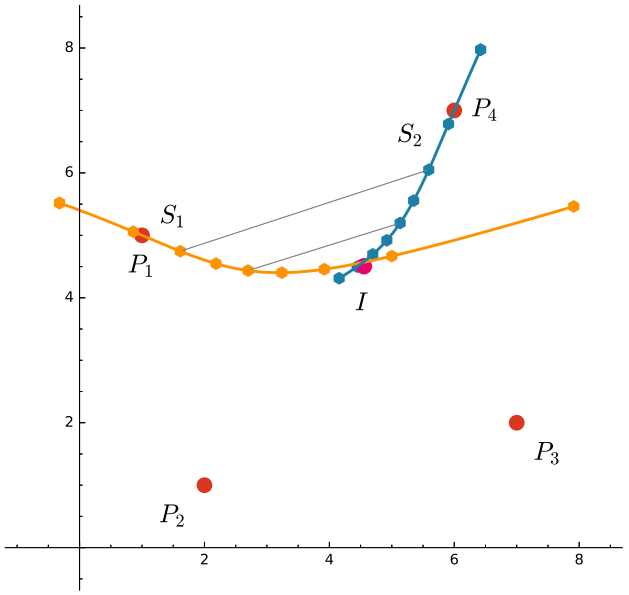


Figure 3: Dynamics of points S_1, S_2 under variation of weight m.

Here

$$X_1 = \frac{1}{30}(45 + 4\sqrt{15}), Y_1 = \frac{1}{30}(90 + \sqrt{15}), r_1 = \frac{2}{15}\sqrt{255}\ ;$$

while the exact expressions for the parameters of C_2 are rather complicated and we present here just only their approximations:

$$X_2 \approx 1.013521,\ Y_2 \approx 8.288416,\ r_2 \approx 5.150241\,.$$

We emphasize that the trajectory of P_3 does not influence the trajectories of S_1 and S_2, i.e. both facilities do not leave the corresponding arcs for any drive of P_3 until the latter reaches the line

$$L \approx \left\{(x, y) \in \mathbb{R}^2 \,\middle|\, y = -1.538431\,x + 10.104975\right\}\,.$$

At this moment, S_1 coincides with S_2 in the point

$$I \approx (3.936925, 4.048287)$$

which yields a solution for the unifacility Weber problem (1) for the terminals $\{P_j\}_{j=1}^4$. Point I is invariant for any position of P_3 in L (Fig. 2).

The scenario for the facilities behaviour in the present example looks similar to the equal weighted case [5], while the problem statement of the next example is of a completely new nature.

EXAMPLE 3. *For the terminals $\{P_j\}_{j=1}^4$ from Example 1 find the loci of facilities S_1, S_2 under the variation of the weight m within $[2, 4.8]$.*

Solution. When the weight m increases, the facilities S_1 and S_2 approach each other along the algebraic curves given in parametric form as $(x_*(m), y_*(m))$ and $(x_{**}(m), y_{**}(m))$ correspondingly. Due to (8), these points collide when m coincides with a zero of the equation $\delta(m) = 0$. The latter can be reduced to an algebraic one

$$24505\,m^{20} - 3675750\,m^{18} + \cdots + 25596924755077 = 0$$

with a zero $m_0 \approx 4.326092$. The collision point

$$I \approx (4.537574, 4.565962)$$

yields a solution for the unifacility Weber problem (1) for the terminals $\{P_j\}_{j=1}^4$.

When m decreases, the facility S_1 moves towards P_1 while S_2 moves towards P_4. The first drive is faster than the second one: S_1 approaches P_1 when m coincides with a zero of the equation $\delta_1(m) = 0$. The latter can be reduced to an algebraic one

$$377145\,m^{12} - 15186678\,m^{10} + \cdots + 8631109474 = 0$$

with a zero $m_1 \approx 2.405703$ (Fig. 3).

4. CONCLUSIONS

The multifacility Weber problem for the case of four terminals and two facilities in the plane has been tackled in its general statement including establishment the conditions for its solubility and deduction the explicit formulae for its solution. The obtained result permits one to analyze the effect of the problem parameter variations to the shape of the network. It also inspires a hope in extensibility of the analytical approach to the problem in its general statement. Indeed, on recalling the idea underlying the proof of Theorem 1, one might expect that the general n-terminal Weber problem can be somehow reduced to a couple of $(n-1)$-terminal problems. Right at the moment, this statement is a mere conjecture, however it is also justified by the known in the literature treatment of the particular case of the problem, namely the Steiner minimal tree problem.

The result can also be useful for the data clusterization problems and for the phylogenetic tree reconstruction [1]. For the latter problem, the extension of the results of present paper to $\mathbb{R}^d, d \geq 3$ is a question for further investigation.

The authors thank the referees for valuable suggestions that helped to improve the quality of the paper.

5. REFERENCES

[1] M. Brazil et al. A novel approach to phylogenetic trees: d-dimensional geometric Steiner trees. *Networks*, 53(2):104–111, 2009.

[2] Z. Drezner et al. The Weber problem. *Springer-Verlag*, Facility Location: Applications and Theory:1–36, 2002.

[3] C. ReVelle and H. Eiselt. Location analysis: A synthesis and survey. *European Journal of Operational Research*, 165(1):1–19, 2005.

[4] A. Uteshev. Analytical solution for the generalized Fermat-Torricelli problem. *Amer. Math. Monthly*, 121(4):318–331, 2014.

[5] A. Uteshev. Some analytics for Steiner minimal tree problem for four terminals, 2015. arXiv:1505.03564v1.

[6] A. Weber. *The theory of the location of industries.* Chicago University Press, Chicago, 1929.

Equations of Dynamics: Using the First Integrals in Numerical Integration

Kirill V. Postnov
St. Petersburg State University
7/9, Universitetskaya nab.
St. Petersburg, 199034, Russia
st016298@student.spbu.ru

ABSTRACT

Equations of dynamics have often the first integrals. In particular, the canonical equations with stationary Hamiltonian have the energy integral. This fact is sometimes used to monitor the accuracy of numerical integration or to construct special methods of numerical integration, focused on the use of such equations.

In this paper we propose a method for correcting the numerical integration at each step based on the known first integrals of these equations.

For the numerical experiments we use three programs: DOP853 (the Dorman-Prince method), ODEX (the Gregg-Bulirsch-Stoer method) and TSMR (explicit method of Taylor series). Programs have been modified so that at each step of numerical integration the corrections ensure the constancy of the first integrals. Numerical experiments show that the proposed method sometimes allows to improve the accuracy of the results on long time intervals. Experiments were carried out on the Two-body problem and the N-body problem (Sun and five outer planets).

Categories and Subject Descriptors

G.1.0 [**Numerical analysis**]: General

General Terms

Theory

Keywords

First integrals, conservative methods, differential equations

1. INTRODUCTION

This study is a logical continuation and propagation of the ideas described in article [8], where the property of conservation of first integrals has been investigated in numerical integration by various methods. The study also proposed a method that facilitates correction of coordinates in each step of integration to improve the integration accuracy. Numerical experiments were conducted on the two-body problem, while four first integrals (one energy integral and three area integrals) were used to correct the coordinates. It was observed that the area integrals are conserved much better than the energy integral. Therefore, we felt it logical to assume that the residual error in energy integral contributes essentially to correction of coordinates.

Besides, investigating only the energy integral allows to make the method more universal and to apply it to a broader class of problems. For example, stationary Hamilton's canonical equations have energy integral.

Based on the aforesaid, we tried to apply the described method using only the energy integral. For numerical experiments, we considered two options for the N-body problem. The first is when $N = 2$ (Sun and Neptune) and the second is when $N = 6$ (Sun and five outer planets). The last one is chosen as the most interesting for researchers [9].

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

2. PROBLEM STATEMENT

We consider a system of differential equations describing the motion of the Solar system planets in heliocentric coordinates [5].

$$\ddot{g}_{ij} = -\gamma(m_0+m_i)\frac{g_{ij}}{r_{0i}^3}+\gamma \sum_{\substack{k\in[1:5]\\k\neq i}} m_s\left[\frac{g_{kj}-g_{ij}}{r_{ki}^3}-\frac{g_{ij}}{r_{0k}^3}\right], \quad (1)$$

where $r_{ij}^2 = \sum_{j\in[1:3]}(g_{ij}-g_{kj})^2$, $r_{si} > 0$, $i \in [1:5]$, $k \in [0:5]$, $k \neq i$, $j = 1,2,3$, γ — universal gravitational constant.

For the two-body problem, these equations take a simpler form [5]:

$$\ddot{g}_j = -\gamma(m_0+m_1)\frac{g_j}{r^3}, \quad (2)$$

where $r = \sqrt{g_1^2+g_2^2+g_3^2}$.

Let us write the same equations in polynomial form. To do this, we use the additional variable method outlined in [3]. This conversion is done with the aim that the TSMR program [5] implementing the Taylor series method [4] is focused on solution of differential equations with polynomial right-hand sides.

$$\begin{cases} \dot{g}_{ij} = p_{ij}, \\ \dot{p}_{ij} = -\gamma(m_0 + m_i)g_{ij}z_{0i}^3 + \\ +\gamma \sum\limits_{\substack{k\in[1:5]\\k\neq i}} m_k\left[(g_{kj} - g_{ij})z_{ki}^3 - g_{ij}z_{0k}^3\right], \\ \dot{z}_{si} = -z_{si}^3 \sum_{j\in[1:3]}(g_{ij} - g_{sj})(p_{ij} - p_{sj}), \end{cases} \tag{3}$$

where $i \in [1:5]$, $s \in [0:5]$, $s < i$, $z_{si} = 1/r_{si}$ [5].

$$\begin{cases} \dot{g}_i = p_i, \\ \dot{p}_i = -\mu g_i z^3, \\ \dot{z} = -z^3(g_1p_1 + g_2p_2 + g_3p_3), \end{cases} \quad i = \overline{1,3}, \tag{4}$$

where $\mu = \gamma(m_0 + m1)$, $z = 1/r$ and γ – universal gravitational constant [5].

Apart from the TSMR program mentioned above, we also used the DOP853 [1] and ODEX [1] programs in our numerical experiments. These two programs implement the Dormand–Prince [7] and Gragg–Bulirsch–Stoer [7] algorithms respectively.

The initial data used in the problems are presented in Table 1. The coordinates are presented in astronomical units

Table 1: Coordinates, velocities and masses of outer Solar System planets

Planet	g_{i1}, g_{i2}, g_{i3}	p_{i1}, p_{i2}, p_{i3}
Jupiter 1047.3486	$0.36484424231671 \times 10^{1}$	$0.51458739160244 \times 10^{-2}$
	$-0.31885628561843 \times 10^{1}$	$0.53770578567746 \times 10^{-2}$
	$-0.14570594138551 \times 10^{1}$	$0.21809890337032 \times 10^{-2}$
Saturn 3497.898	$0.860812008268 \times 10^{-1}$	$-0.58771844012993 \times 10^{-2}$
	$0.83323915033502 \times 10^{1}$	$-0.48077953809 \times 10^{-4}$
	$0.34416852468077 \times 10^{1}$	$0.0234400362575 2 \times 10^{-2}$
Uranus 22902.98	$-0.16894580321104 \times 10^{2}$	$0.15371000247255 \times 10^{-2}$
	$-0.6802790274465 \times 10^{1}$	$-0.3460060959201 \times 10^{-2}$
	$-0.2742015202544 \times 10^{-1}$	$-0.15378414591329 \times 10^{-2}$
Neptune 19412.24	$-0.1196586398024 \times 10^{2}$	$0.28675086989466 \times 10^{-2}$
	$-0.25873934395511 \times 10^{2}$	$-0.10995281783636 \times 10^{-2}$
	$-0.10297822028119 \times 10^{2}$	$-0.5224555223166 \times 10^{-3}$
Pluto 1.35×10^{8}	$-0.29628255336199 \times 10^{2}$	$0.747289767668 \times 10^{-3}$
	$-0.5542237950558 \times 10^{1}$	$-0.30948199515322 \times 10^{-2}$
	$0.7229040993881 \times 10^{1}$	$-0.12025228562757 \times 10^{-2}$

(AU), while the speed is presented in astronomical units per day (AUd^{-1}) (mean solar day). To make computations easier the masses of the planets relative to the Sun are used in the programs instead of the true masses.

It is assumed that the initial data recorded in Table 1 are defined absolutely accurate.

We write the analytical formulas of energy integrals. Energy integral in the two-body problem [6]:

$$H(g,p) = \frac{1}{2}\left(p_1^2 + p_2^2 + p_3^2\right) - \frac{\mu}{\sqrt{g_1^2 + g_2^2 + g_3^2}}. \tag{5}$$

Energy integral in the N-body problem [6]:

$$H(g,p) = \frac{1}{2}\sum_{i=1}^{n-1} m_i\left(p_{i1}^2 + p_{i2}^2 + p_{i3}^2\right) - \frac{1}{2m}\left[\left(\sum_{i=1}^{n-1} m_i p_{i1}\right)^2 + \right.$$
$$\left. + \left(\sum_{i=1}^{n-1} m_i p_{i2}\right)^2 + \left(\sum_{i=1}^{n-1} m_i p_{i3}\right)^2\right] - \gamma \sum_{i<j,j\in[1:n-1]} \frac{m_i m_j}{r_{ij}}, \tag{6}$$

where $m = \sum_{l=0}^{n} m_l$, $r_{ij} = \sqrt{\sum_{j\in[1:3]}(g_{ij} - g_{kj})^2}$.

3. INTEGRAL CONSERVATION METHOD

Below is a description of the method used to conserve the values of the first integrals over an entire integration interval. This method is based on the method described in [8] view of the fact that we try to conserve only the energy integral.

Let $x_0 = (g_0, p_0)$ be the initial data of the Cauchy problem for the system (1) or (2). Let $x = (g, p)$, $\dim(g, p) = n$ be the value on the k-th step of integration. The function $H(x(t)) = C_0 = \text{const}$ is the integral energy of the corresponding system.

Let us denote by $C_1 = H(x_1)$ the energy integral values in the k-th step. We decompose the integral in a Taylor series by discarding all the higher order terms

$$H(x + \delta) = C_{i0} = H(x) + \sum_{k=1}^{n} \frac{\partial H(x)}{\partial x_k}\delta_k.$$

Introducing the denotation $d = C_0 - C_1$, we get:

$$\sum_{k=1}^{n} \frac{\partial H(x)}{\partial x_k}\delta_k = d.$$

Let's seek $\delta_1, \ldots, \delta_n$ such that $\sum_{k=1}^{n} \delta_k^2 \to min$. To find them, we will use the method of Lagrange multipliers:

$$L(\delta, \lambda) = \sum_{i=1}^{n} \delta_i^2 + \lambda\left(\sum_{k=i}^{n} \frac{\partial H(x)}{\partial x_i}\delta_i - d\right), \tag{7}$$

$$\frac{\partial L(\delta,\lambda)}{\partial \delta_i} = 2\delta_i + \frac{\partial H(x)}{\partial x_i}\lambda = 0, \quad i = \overline{1,n}, \tag{8}$$

$$\frac{\partial L(\delta,\lambda)}{\partial \lambda} = \frac{\partial L(\delta,\lambda)}{\partial \lambda} = \sum_{k=i}^{n} \frac{\partial H(x)}{\partial x_i}\delta_i - d = 0. \tag{9}$$

From (8), δ_i becomes:

$$\delta_i = -\frac{1}{2}\frac{\partial H(x)}{\partial x_i}\lambda = 0, \quad i = \overline{1,n}. \tag{10}$$

Substituting (10) in (9) we obtain:

$$\lambda = -\frac{2d}{\sum_{i=1}^{n}\left(\frac{\partial H(x)}{\partial x_i}\right)^2} = 0. \tag{11}$$

The final formula for computing the numerical value of corrections is as follows:

$$\delta_i = \frac{d \cdot \frac{\partial H(x)}{\partial x_i}}{\sum\limits_{k=1}^{n}\left(\frac{\partial H(x)}{\partial x_k}\right)^2}. \tag{12}$$

4. NUMERICAL EXPERIMENTS

Taking into account the method described above, we modified all the programs in such a way as to correct the solution values using this method at the end of each period. These modifications are presented in Fig. 1

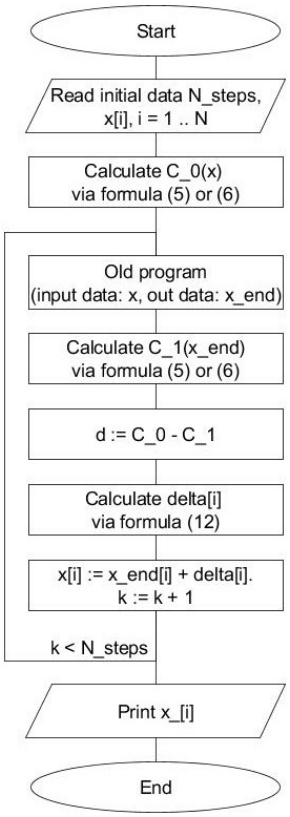


Figure 1: Block diagram of re-engineered program

4.1 Two-body problem

The numerical experiment results are presented in Fig. 2.

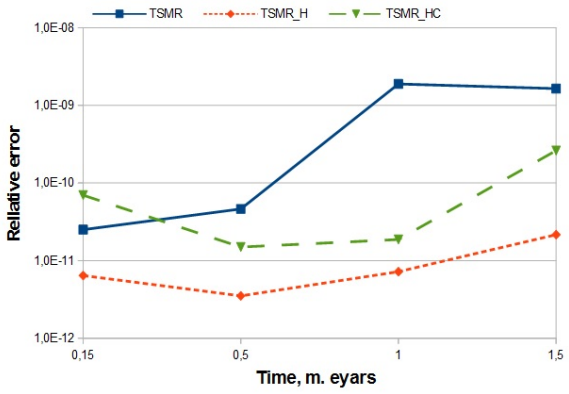


Figure 2: Results of integration using the TSMR program

The figure shows the results of integration carried out using the TSMR program and its modifications. TSMR-HC is the modification of TSMR and it conserves the energy integral and the area integral, while TSMR-H is the modification that conserves only the energy integral.

It can be seen in the graph that among all the modifications, the TSMR-H program yields the best results (1-2 orders of magnitude). One can note that this result is remarkable also by the fact that integration accuracy improved despite reducing the amount of integrals conserved. Besides, as has been noted above, this simplifies the program itself and makes it more versatile.

The DOP853 and ODEX programs give similar results. Detailed graphs and tables for a variety of programs can be found in [2].

4.2 Problem of movement of outer Solar System planets

Given the results in the preceding paragraph, we have attempted to test our program on the six-body problem. Here, we decided to use the TSMR-H modification that conserves only the energy integral. The results obtained were unpleasantly surprising.

Since the N-body problem has no periodic solution for $N > 2$, round-trip integration was therefore used to assess the relative solution errors: the equations are first integrated from point 0 to point T, and then, taking the data obtained as initial data, the equations are integrated from point T to point 0.

As shown by experiments on all the tested intervals, the integration accuracy worsens by several orders of magnitude. After obtaining these results, we decided to check how the modified program behaves in one step for round-trip integration.

The results are shown in Table 2.

Table 2: Result of integration in one step

TSMR	TSMR-H	TSMR-H-NEW
1.597274×10^{-12}	$6.743340015638 \times 10^{-3}$	$6.743340015644 \times 10^{-3}$

The Table 2 contains errors for integration using TSMR and its two modifications. The TSMR-H modification differs from TSMR-H-NEW with the fact that in the TSMR-H program, the initial value of energy integral that we conserve further is calculated once at the starting point of integration and is used in the entire round-trip integration area. In the TSMR-H-NEW program, the initial value is calculated twice – the first time is at the starting point of integration when integrating forward and the second time is at the starting point of integration when integrating back.

As it follows from the table, although the values of the first integrals are conserved in both cases, integration accuracy does not improve but worsens.

5. CONCLUSIONS

The paper describes an integration method that allows to conserve the energy integral value in the entire integration interval. This method was used to modify some existing programs and conduct numerical experiments.

In the two-body problem, using the proposed method can improve integration accuracy by $1 - 2$ orders of magnitude.

In the problem of five outer Solar System planets, applying the method reduces the integration accuracy. Character of

result degradation is not clear yet. We can note the following reasons:

- instability of the 6-body problem. In other words, a small change of initial data influences on further motion significantly.
- unintentional mistake in the program modification. Although there is not any explicit or indirect confirmation of this fact (energy integrals are saved with high accuracy, modification does not influence on main integration functions Fig.1), we do not eliminate this possibility.

6. REFERENCES

[1] Fortran codes, June 2016. url: http://www.unige.ch/ hairer/software.html.

[2] Postnov's scientific papers, June 2016. url: http://www.postnov.pro/science/articles/ (in Russian).

[3] L. Babadzhanjanz. The additional variables method. *Vestnik Sankt-Peterburgskogo Universiteta.[Vestnik of St. Petersburg State University]*, 10(1):3–11, March 2010 (in Russian).

[4] L. Babadzhanjanz. The taylor series method. *Vestnik Sankt-Peterburgskogo Universiteta.[Vestnik of St. Petersburg State University]*, 10(3):13–29, September 2010 (in Russian).

[5] L. Babadzhanjanz and A. Bolshkov. Implementation of the taylor series method for solving ordinary differential equations. *Vychislitel'nye Metody i Programmirovanie[Numerical methods and Programming]*, 13:497–510, 2012 (in Russian).

[6] G. N. Duboshin, editor. *Spravochnoe rukovodstvo po nebesnoj mehanike i astrodinamike [Reference guide celestial mechanics and astrodynamics]*. Nauka, Moscow, 1976 (in Russian).

[7] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I (2Nd Revised. Ed.): Nonstiff Problems*. Springer-Verlag New York, Inc., New York, NY, USA, 1993.

[8] K. V. Postnov. Using the first integrals to increase the precision of ODEs numerical integration. *Control Processes and Stability*, 3(1):230–234, April 2016 (in Russian).

[9] A. Roy, I. Walker, A. Macdonald, I. Williams, K. Fox, C. Murray, A. Milani, A. Nobili, P. Message, A. Sinclair, and M. Carpino. Project longstop. *Vistas in Astronomy*, 32:95 – 116, 1988.

Design of a Test System for Analog and Mixed-Signal Electronic Devices

Alexander Vasiliev
Peter the Great St. Petersburg Polytechnic University
29 Polytechnicheskaya st.
195251 St. Petersburg Russia
alex.m.vasil@gmail.com

Yuri Kotchetkov
Peter the Great St. Petersburg Polytechnic University
29 Polytechnicheskaya st.
195251 St. Petersburg Russia
genesis91@yandex.ru

ABSTRACT

This paper presents test and diagnostic system for analog and mixed-signal electronic devices. System is based on single-board instrumentation platform and includes custom designed programmable power supplies. Test system software enables rapid creation of automated workstations for various electronic devices. Development of test system software is described. Examples of automated workstations based on this system for existing analog devices are given.

Categories and Subject Descriptors

B.8.1 [**Hardware**]: Performance and Reliability—*reliability, testing, and fault-tolerance*; J.2 [**Computer Applications**]: Physical Sciences and Engineering—*electronics*

Keywords

Automated test system, analog and mixed-signal devices, automated workstation, Red Pitaya

1. INTRODUCTION

Nowadays, production of analog and mixed-signal electronic modules for severe climatic and radiological conditions is increasing. Because of strict requirements for their reliability and huge number of laser-trimmed and adjusted components, those devices undergo numerous iterations of functional testing and parametric measurement [2]. At that, given procedures are often performed by hand with a great number of standalone instruments, which demands high qualification of operating personnel. For that reason, test automation of those electronic devices is of immediate interest.

Electronic enterprises produce vast variety of such devices, each of them in quantities of several thousand a year. That is why, it is more feasible to develop automated test systems of modest cost (within $10,000). It is desirable that these systems are mobile and do not demand highly qualified operating personnel. The most effective way is to develop several automated test system based on a universal instrumentation platform.

We developed a test and diagnostic system which contains all necessary measuring instruments and control software for rapid development of automated test systems for analog and mixed-signal devices. This system is intended for testing various electronic devices and can be operated by low-qualified staff. We have made two KDK-MT hardware units (revision 2 unit is shown in Fig. 1). We also developed several automated workstations based on KDK-MT platform and evaluated our test system hardware and software by performing functional test of various analog electronic devices.

The rest of this paper is organized as follows. Section 2 reviews existing test systems for analog and mixed-signal devices and also presents universal single-board instrumentation platform Red Pitaya. Section 3 describes hardware of test system. In section 4 we present development of our system software. Finally, section 5 proposes our future ways to deal in this area and makes a conclusion.

2. REVIEW

There are a lot of systems for automated test of analog and mixed-signal electronic devices. Test systems of a common type consist of standalone measuring instruments, which have a PC-interface for automatic control. Those systems often use LabView software. More advanced approach is to use instruments based on modular instrumentation platforms PXI, VXI and LXI [6]. These platforms enable creating test systems for almost any electronic device. The main disadvantage of those systems is their high cost, often connected with excessive for test problems accuracy. Another drawback is low mobility. Besides that, setup of such systems demands highly qualified personnel.

A new approach to build up test systems is to use single-board instrumentation and control platforms. Such devices contain analog front-end (oscilloscopes, arbitrary waveform generators) and a computer to perform instruments control, result processing and communication with user. One of such devices is Red Pitaya board [4]. It appeared on the marked in 2014.

Red Pitaya has two fast ADC and DAC channels with maximum sampling rate of 125 MSPS, 16 GPIO, additional ADC and DAC with lower sampling rate. The board is equipped with a Xilinx Zynq-7010 SoC. It contains dual core ARM processor and FPGA with 28k logic cells. The operating system is an embedded version of Debian 8 Jessie. A

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

Table 1: Comparison of test systems

Features	NI VirtualBench	KDK-MT
Oscilloscope, sampling rate	2 ch. 1 GSPS	4 ch. 125 MSPS
Generator sampling rate	1 ch. 1 GSPS	4 ch. 125 MSPS
DC voltmeter	multimeter 5 1/2 digits	12-bit auxiliary ADC
Digital I/O	8	16
Power supplies	1x 0..6 V 1x 0..25 V 1x -25..0 V	4x isolated 3..20 V 14x fixed voltage 3.3 V to 24 V
Price of hardware unit	$4000	est. $2000

user's computer can be connected via Ethernet. The main feature of Red Pitaya architecture is that all measuring instruments control logic is implemented in FPGA. This enables board customization for specific application [1].

In Table 1 our test system is compared to another single-unit instrument - National Instruments VirtualBench [3]. This is a universal single-unit test and measurement system which combines oscilloscope, signal generator, multimeter, digital I/O and power supplies.

The main disadvantage of NI VirtualBench as a universal platform for test systems (automated workstations) is low number of generator and oscilloscope channels; however, this system has higher sampling rate of analog front-end which may be useful for testing some devices. In addition, KDK-MT has more power supplies and four of them are isolated. A large number of measurement channels and power supplies are essential for development of automated test systems for vast variety of analog electronic devices.

3. DESIGN OF TEST SYSTEM

Fig.1 presents KDK-MT test system. It contains four oscilloscope channels and four arbitrary waveform generators with 50 MHz bandwidth. Digital interface consists of 14 GPIO. Besides that, there are four programmable power supplies with a useful current measurement function and 14 fixed voltage supplies. This system is designed to test devices with supply voltage up to 20 V and operating frequency up to 50 MHz.

The KDK-MT system is based on two universal instrumentation boards Red Pitaya. Boards and user PC are connected in network using Ethernet switch. One of the boards also acts as a primary controller of the test system.

Fig. 2 presents a custom developed programmable isolated power supply for KDK-MT, which can measure current consumption within the accuracy of 0.1 mA. This function is crucial for the diagnostic method of analog devices basing on current consumption, which was proposed by our research group. Power supply is based on STM8L 8-bit microcontroller which features 1 MSPS 12-bit ADC. Programmable power supplies are connected to master Red Pitaya board via UART interface.

Testing of analog devices often includes measuring of pulse response, which requires advanced signal acquisition synchronization. Usually, in this case signal generator and oscilloscope must be triggered simultaneously. KDK-MT in-

Figure 1: Test and diagnostic system KDK-MT.

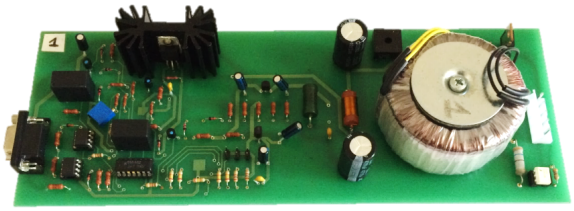


Figure 2: Programmable power supply for KDK-MT.

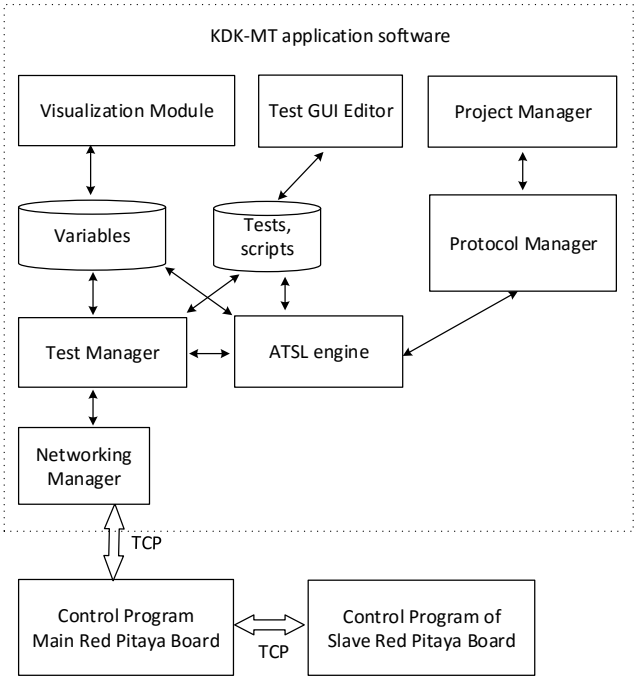


Figure 3: Architecture of test system software.

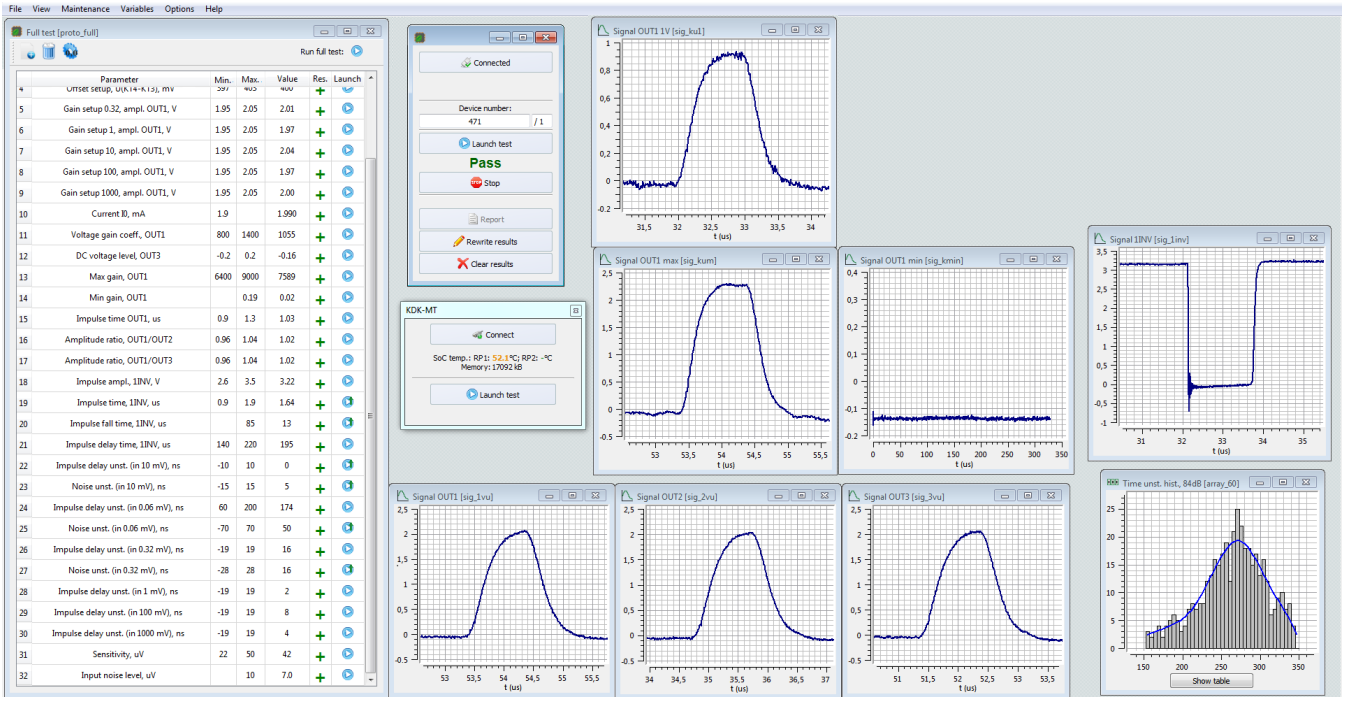


Figure 4: User interface of automated workstation for impulse amplifier testing.

ternal synchronization system allows oscilloscopes of both Red Pitaya boards to be triggered by synchronization signal from any generator channel. This signal is emitted in the beginning of every generator waveform period. To implement this function we modified the default configuration of FPGA. Trigger signal from generator control logic in FPGA is routed to the special pin on Red Pitaya expansion connector. These pins of both boards are connected to external trigger inputs of master and slave Red Pitaya by synchronization bus.

4. TEST SYSTEM SOFTWARE

Software of test system consists of control program of master and slave Red Pitaya boards and a client application. Control program is designed to interchange data between measurement equipment and digital interfaces of Red Pitaya boards and to interact with client program on user's PC.

4.1 Control program

Control program is running on the master Red Pitaya board. The slave board is controlled by a modified version of this program. It comprises only modules of oscilloscopes, generators and digital outputs management and a networking module which enables interaction with master board. Architecture of system software is shown in Fig. 3. Most of the test data, including KDK-MT generated and acquired signals, as well as measurement processing results, is stored in generalized container objects called variables. Test system scripting engine can read and modify these variables, enabling flexible control of testing process.

The control logic of board oscilloscopes and generators is implemented in FPGA. Data buffers and configuration registers are mapped into RAM with specific physical addresses. Control program communicates with instruments through reading and writing these registers in memory [5]. Digital outputs are also controlled that way.

4.2 Client program

User is working with KDK-MT test system using client application. This software is an integrated environment for test creation, tuning and launching. After completion of test its results are displayed as tables and charts. Program uses multiple document interface to allow customization of client application GUI for particular device test.

KDK-MT is used as a platform for development of automated workstation for particular analog and mixed-signal devices. An automated workstation consists of KDK-MT test system, an electronic coupling device for signal multiplexing between test system and device under test, and a client application which runs on user's PC.

Client application operates in two modes:

- *Test program setup mode.* In this mode user creates sequential description of test, which is named test protocol, and designs graphical interface of automated workstation. GUI includes protocol as a table with test stages, launch buttons and result marks; plots and information panels. This mode can also be used for analysis of device under test functioning. In setup mode a highly qualified operator is required.

- *Automated test mode.* A low-skilled user can operate in this mode. Operator would be responsible for connecting device to automated workstation coupling device, launching given test program and analyzing test results from the final protocol. Sample user interface of automated workstation is shown in Fig. 4.

System software (control and client programs) is written in C++ language using Qt library. Key features of Qt are

signal and slot mechanism which enables convenient handling of asynchronous events; TCP sockets (using signals and slots), XML parser (this format is used to store test results); container classes and etc. Client application uses Qt as graphical framework. The Qwt library is used for displaying plots.

Test programs are written in scripting language ATSL. This language has simple syntax, close to numerical computing systems Matlab or Octave. It contains basic set of arithmetic and logic operators, cycles and conditional statements, as well as special purpose directives to control testing. A key feature of complex software is capability to upload a part of the script for execution in control program. At that, constant intensive network communication between KDK-MT and client application during test is eliminated. This results in a significant increase of test performance. Script uploading is important when a repetitive measurement of single parameter is required.

After completion of test final protocol is automatically filled up and can be printed or exported into PDF format. User can customize format of this document. We employ free HTML editor CKEditor for this purpose. It runs on Web engine QWebKit, which is part of Qt framework.

4.3 Automated workstations

We developed several automated workstations based on KDK-MT test system for existing analog electronic devices. First one is performing parametric measurement of special-purpose impulse amplifier with high sensitivity. The second workstation is designed for automated measurement of DC current gain of bipolar transistors and grouping them into pairs. The last one performs functional test of Manchester code transceiver.

For each automated workstation we created a coupling device in which the device under test is plugged in. These coupling devices usually contain multiplexing switches, amplifiers and other circuits required for a particular test object. A single KDK-MT unit can be used in several automated workstation; in this case switching between them requires only connecting cables and opening a test project in client application.

5. CONCLUSION

In this paper we presented the KDK-MT test system for analog and mixed-signal electronic devices. We described design of software, which enables rapid development of new automated workstations for vast variety of electronic devices. This system is cheaper than others presented on the market. Besides, automated workstation based on KDK-MT can be operated by low-qualified personnel.

Future work includes design of programmable power supply with improved characteristics, essential for current consumption diagnostic methods. Also we are planning to use JavaScript as a test scripting language instead of ATSL because of its higher performance and bigger capabilities, which are required for complicated processing of test results.

6. REFERENCES

[1] A. C. Cardenas-Olaya, S. Micalizio, M. Ortolano, C. Calosso, E. Rubiola, and J.-M. Friedt. Digital electronics based on Red Pitaya platform for coherent fiber links. In *Proc. European Frequency and Time Forum (EFTF)*. pages 1-4. IEEE, 2016.

[2] J. Cressler, H. Mantooth. *Extreme Environment Electronics*. CRC Press, 2012.

[3] NI VB-8012 Device specifications: 2016. http://www.ni.com/pdf/manuals/371527d.pdf. [Accessed: June 2016].

[4] B. Landoni. Red Pitaya: the open-source electronic laboratory: 2014. http://www.open-electronics.org/red-pitaya-the-opensource-electronic-laboratory/. [Accessed: June 2016].

[5] Red Pitaya FPGA memory map: 2015. http://wiki.redpitaya.com/tmp/RedPitaya_HDL_memory_map.pdf [Accessed: June 2016].

[6] What is LXI?: 2015. http://www.ni.com/white-paper/7255/en/. [Accessed: June 2016].

Design of an Asynchronous Inverse Discrete Cosine Transform Circuit on an FPGA

Taiki Urakawa
University of Aizu
Aizuwakamatsu 965-8580, Japan
m5201140@u-aizu.ac.jp

Hiroshi Saito
University of Aizu
Aizuwakamatsu 965-8580, Japan
hiroshis@u-aizu.ac.jp

ABSTRACT

In this paper, we design a low power asynchronous Inverse Discrete Cosine Transform (IDCT) circuit on a Field Programmable Gate Array (FPGA). We synthesize the asynchronous IDCT circuit assigning the maximum delay constraints for data-paths. In the experiment, we evaluate the area, execution time, power consumption, and energy consumption of the designed asynchronous IDCT circuit. The power was reduced 24 % compared to the synchronous counterpart.

Categories and Subject Descriptors

B.6.0 [**LOGIC DESIGN**]: General

General Terms

Design

Keywords

Asynchronous circuits, FPGA, IDCT

1. INTRODUCTION

Recently, importance of image compression is increasing to save memories on portable devices. As portable devices are battery drive, low power image compression circuits are required for battery saving as mentioned in [1, 2, 3].

Inverse Discrete Cosine Transform (IDCT) is used in many multimedia applications such as JPEG [4] and MPEG [5]. IDCT is one of the heaviest processes in image compression. In IDCT, information in frequency domain is converted to information in time-space domain. As high performance and low power are required to IDCT, IDCT is usually realized as a specialized circuit.

Most of circuits including IDCT circuit are a synchronous circuit. Circuit components in synchronous circuits are controlled by global clock signals. Synchronization failure by clock skew and power consumption on the clock network will be significant in synchronous circuits when the semiconductor submicron technology is advanced more and more.

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

In asynchronous circuits, circuit components are controlled by local handshake signals. Due to the absence of global clock signals, there is no problem related to clock signals. In addition, asynchronous circuits are potentially low power consumption and low electromagnetic radiation compared to synchronous circuits. However, the design of asynchronous circuits is difficult more than the design of synchronous circuits due to the absence of global clock signals.

Recently, Field Programmable Gate Arrays (FPGAs) are well used in embedded systems. This is because FPGAs are long lifetime and low design cost. As designers can change circuit structure freely, FPGAs are adaptive for specification change and extension.

In [6], an asynchronous reconfigurable architecture was proposed for voice processing and image processing. This work used a special gate-level library for the design. It reported latency, throughput, and circuit area. However, the effectiveness is not well described because there is no comparative evaluation.

In this work, we design an asynchronous IDCT circuit. The asynchronous circuit is based on bundled-data implementation. After the design, we evaluate the designed asynchronous IDCT circuit comparing with the synchronous counterpart.

The rest of this paper is organized as follows. In section 2, we describe background used in this paper. In section 3, we describe the design of an asynchronous IDCT circuit. In section 4, we describe the experimental result. In section 5, we conclude this paper.

2. BACKGROUND

2.1 Asynchronous Circuits with Bundled-data Implementation

The bundled-data implementation is one of implementation methods of asynchronous circuits. It uses N+2 signals to represent N bit data. Additional 2 signals are request signal and acknowledge signal. In bundled-data implementations, the completion of operations are guaranteed by delay elements which are inserted on request signals. Therefore, the performance of bundled-data implementations depends on the delay of the control circuit with delay elements.

Figure 1 shows a circuit model of bundled-data implementation used in this paper. The right side of Fig. 1 is a data-path circuit and the left side is a control circuit.

The data-path circuit consists of registers (reg_k) ($0 \leq k \leq e$), functional units (fu), glue logics (g), and multiplexers (mux). They are the same as the ones used in synchronous circuits. In addition, the data-path circuit includes delay elements hd_k to satisfy hold constraints. Initially, hd_k is

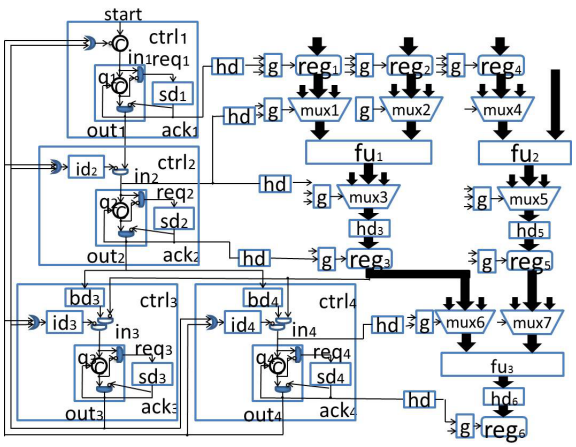


Figure 1: Asynchronous circuit with bundled-data implementation.

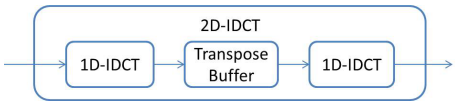


Figure 2: Structure of 2D-IDCT.

just a wire from the input to the output. Glue logics receive signals from the control circuits and generate control signals for functional units, multiplexers, and registers.

The control circuit consists of control modules $ctrl_i$ $(1 \leq i \leq n)$. A control module controls one state and consists of Q-module q_i, glue logic (g), three delay elements id_i, sd_i and bd_i, and C-element c_i. id_i is used to satisfy idle constraints. sd_i is used to satisfy setup constraints. bd_i is used to satisfy branch constraints. C-element c_i is a synchronization component of the input signals. The output of C-elements becomes 0 when all inputs are 0, it becomes 1 when all inputs are 1. Otherwise, the output is not changed.

The behaviors of bundled-data implementations are as follows. In this paper, $signal+$ represents a positive edge of $signal$, $signal-$ represents a negative edge of $signal$. This circuit model starts by $start+$ from outside. A control module $ctrl_i$ starts by $out_{i-1}+$ from $ctrl_{i-1}$. The control signals of functional units and multiplexers in the data-path circuit are generated by in_i+ through glue logics. q_i generates req_i+ by in_i+. req_i+ goes back to q_i through sd_i+ as ack_i+. q_i generates req_i- then goes back to q_i as ack_i-. After this, data are written into register by ack_i-. Finally, q_i generates out_i+ by ack_i- and moves the control to the next control module. After the last control module generates out_i+, all control modules generate in_i- and out_i-.

There are 4 types of timing constraints in asynchronous circuits with bundled-data implementation used in this paper. These are described in [7]. If some of constraints are not satisfied, we need to satisfy them adjusting delay elements sd_i, hd_k, bd_i, and id_i.

2.2 IDCT

Inverse Discrete Cosine Transform (IDCT) transforms signals from time-space domain into frequency domain. 2D-IDCT consists of two 1D-IDCT as Figure 2. The first 1D-IDCT processes row information and the last 1D-IDCT processes column information. We represent N-unit input data of the frequency domain as $(F_0, \cdots, F_{N-1})$, and N-unit output data of the space-time domain as $S(k)\,(k = 0, \cdots, N-1)$. The formula of IDCT is represented as (1).

$$S(k) = \frac{1}{2}F_0 + \sum_{n=1}^{N-1} F_n \cos\left(\frac{\pi}{N} n \left(k + \frac{1}{2}\right)\right) \quad (1)$$

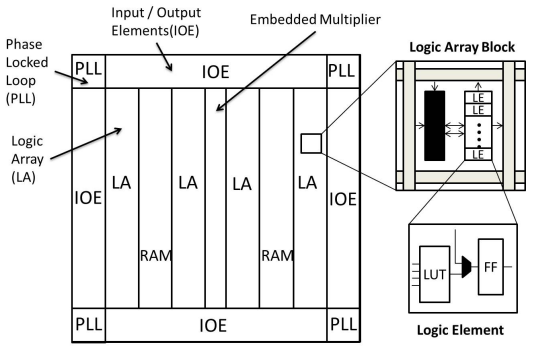


Figure 3: Structure of Altera Cyclone IV [8].

2.3 FPGA

Field Programmable Gate Array (FPGA) is a reconfigurable device where designer can change circuit structure freely. Therefore, FPGA is adaptive with specification change and expansion. The demands of FPGAs in embedded systems are increased because of its low design cost.

In this work, we implement an asynchronous IDCT circuit on an Altera FPGA. Figure 3 represents the structure of Altera Cyclone IV FPGA. It consists of Phase Locked Loops (PLLs), Input/output Elements (IOEs), Embedded Multipliers (EMs), Random Access Memories (RAMs), and Logic Arrays (LAs). PLLs are used for multiplication and division of clock signals, IOEs are used for inputs and outputs to outside. EMs are high performance multipliers and RAMs are memory blocks.

3. DESIGN OF AN ASYNCHRONOUS IDCT CIRCUIT

In this work, we design a low power asynchronous IDCT circuit on an FPGA. As power consumption depends on the execution time, we assign the maximum delay constraints for data-paths related to setup constraints. In this section, we describe the design of the circuit model, the generation of the maximum delay constraints, and the design flow.

3.1 Design of an Asynchronous IDCT Circuit Model

The data-path circuit of the asynchronous IDCT used in this work is the same as the data-path circuit of a synchronous IDCT. The control circuit is implemented by assigning a control module $ctrl_i$ to each state after making state transition diagram referring to the synchronous IDCT circuit.

Figure 4 shows the structure of the control module $ctrl_i$ for Altera Cyclone IV. It includes primitives DLATCH, LCELL, and AND2 of Altera Cyclone IV. Two dummy modules are inserted to assign wires as through points for path delay analysis. Dummy modules are just a wire. We insert two

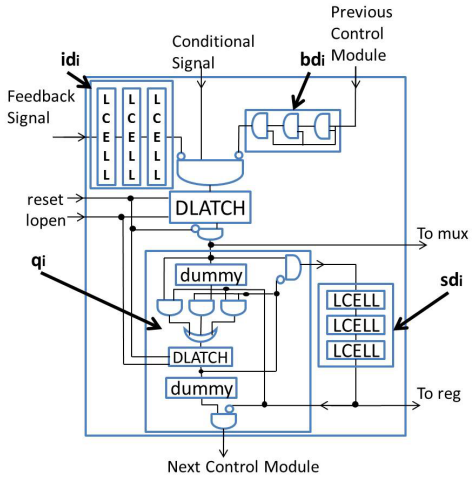


Figure 4: Structure of control module $ctrl_i$.

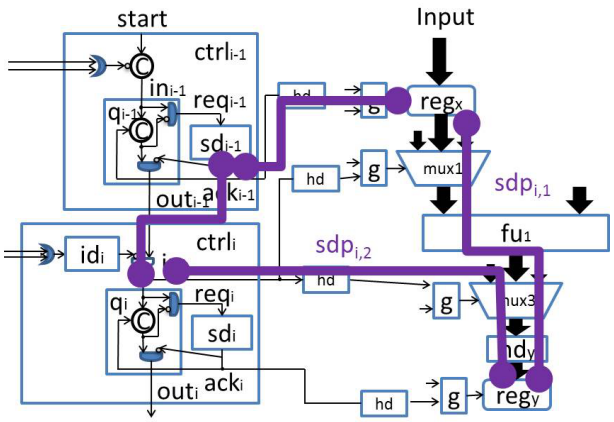


Figure 5: Data-paths to assign the maximum delay constraints.

DLATCHes to analyze path delays correctly. The delay elements id_i and sd_i consist of LCELLs which work as buffers. The delay elements bd_i consist of AND primitives. We put "synthesis keep" attribute for the output pin of sd_i to analyze path delays correctly. Moreover, to avoid optimization for control modules, we assign "design partition" attribute for control modules. Finally, we represent the asynchronous IDCT circuit model using Verilog HDL.

3.2 Generation of the Maximum Delay Constraints and Setting of Margins

3.2.1 *Generation of the Maximum Delay Constraints*

We assign the maximum delay constraints to data-paths $sdp_{i,p}$ related to setup constraints. This is because we would like to design a bundled-data implementation which satisfies a given latency constraint.

Figure 5 represents two data-paths $sdp_{i,1}$ and $sdp_{i,2}$. A data-path $sdp_{i,p}$ is divided into 2 sub-paths. The former is from sd_{i-1} to source register reg_x or from sd_{i-1} to DLATCH before q_i. The latter is from reg_x to destination register reg_y or DLATCH to reg_y.

```
set_max_delay
  -from ctrl0|sd0|out
  -to rr0|out[*]
  1.12
```

Figure 6: Example of set_max_delay command.

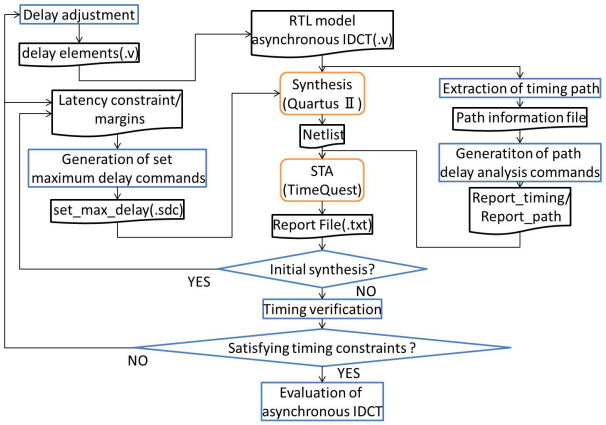


Figure 7: Design flow.

To decide the values of the maximum delay constraints, we use the calculation method in [7]. Figure 6 represents an example of the generated maximum delay constraints using "set_max_delay" command.

3.2.2 *Setting of Margins*

For data-path delays and control path delays, we setup margins. Uncertain effects to data-path delays after FPGA implementation may exist. Therefore, we setup a margin for data-paths. The margin is decided by a given latency constraint L and actual path delays on the target FPGA. If timing violations exist during simulation after the setup of the margin, we increase the margin to solve timing violations.

In bundled-data implementation, from setup constraints, the minimum delay of a control path $scp_{i,p}$ must be larger than the maximum delay of a data-path $sdp_{i,p}$. However, if the minimum delay of $scp_{i,p}$ is very large for the maximum delay of $sdp_{i,p}$, it may results in performance degradation. In such a case, we need to reduce primitives from delay elements. On the other hand, the reduction of primitives from delay elements increases the number of delay adjustments caused by timing violations. Therefore, we setup a margin for control paths. If the minimum delay of $scp_{i,p}$ overs the margin, we remove primitives from corresponding delay elements. The value of the margin is increased if the number of delay adjustments overs a threshold value

3.3 Desing Flow

Figure 7 shows a design flow used in this work. The inputs of this flow are a bundled-data implementation model represented by Verilog HDL (explained in section 3.1), a latency constraint L, and margins for data-path delays and control path delays.

First, we extract paths related to timing constraints from

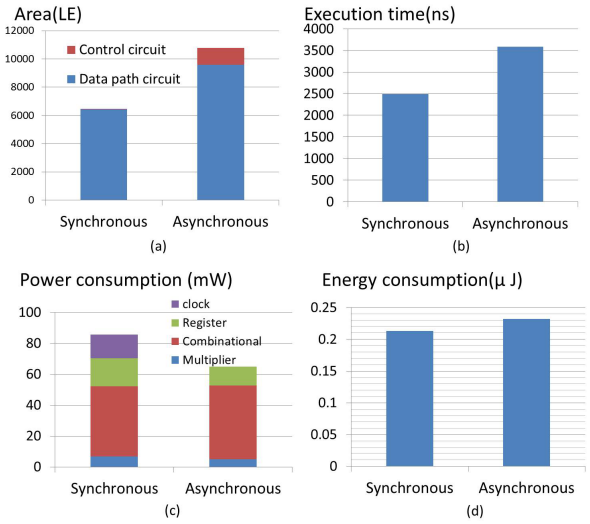


Figure 8: Experimental result: (a) area, (b) execution time, (c) dynamic power, and (d) energy consumption.

the bundled-data implementation model. The extracted information is represented in a path information file. From the path information file, we generate "report_timing" and "report_path" commands which are commands to analyze path delays using a timing analyzer. If the end point of a path is a register, we use "report_timing" to analyze the setup or hold time of the register. Otherwise, we use "report_path" command. Through the synthesis and static timing analysis using Altera Quartus II and TimeQuest, the flow is branched. If it is the initial synthesis, we generate the maximum delay constraints for data-paths related to setup constraints explained in section 3.2. If it is not the initial synthesis, we verify 4 types of timing constraints described in [7] manually. If all timing constraints are satisfied, we finish the design. Otherwise, we carry out delay adjustment to satisfy timing constraints. Synthesis is repeatedly carried out until all timing constraints are satisfied. If the number of timing constraints over a threshold value, we increase the margin for control path delays and repeat synthesis.

4. EXPERIMENTS

In the experiment, we evaluate the designed asynchronous IDCT circuit in terms of area, execution time, dynamic power consumption, and energy consumption comparing with a synchronous counterpart. We use Altera Quartus II ver.14.1 and ModelSim-Altera Starter Edition ver.10.3 for synthesis and simulation. The target device is Altera Cyclone IV (EP4CE115F29C7).

Initially, we explore the fastest synchronous IDCT circuit in clock frequency by changing clock cycle time. The clock cycle time of the fastest synchronous IDCT is 14 ns. The latency constraint for the asynchronous IDCT circuit is set to 308 ns. The margin for data-path delays is set to 1.4 ns (10% of clock cycle time) and the margin for control path delays is 5.6 ns (40% of clock cycle time).

Figure 8 (a) represents the area of the designed IDCT circuit in terms of logic elements reported by Quartus II. The area overhead of the asynchronous IDCT circuit is 33 % compared to the synchronous IDCT circuit. The overhead is caused by the area of control modules and delay elements. Figure 8 (b) represents the execution time when an arbitrary test input is given and simulated using ModelSim-Altera. The overhead of the execution in the asynchronous IDCT circuit is 44 % compared to the synchronous IDCT circuit. This overhead comes from the margin for control path delays. As we set a large enough value (5.6ns), more LCELLS are inserted to delay elements. Figure 8 .(c) represents the dynamic power consumption reported by PowerPlay Power Analyzer in Quartus II when the designed circuit and simulation result are given. We can reduce 24% of dynamic power consumption due to the reduction of the power consumption for clock network and register. Figure 8 .(d) represents the energy consumption obtained by the product of the dynamic power consumption and the execution time. It is increased 9% due to the increase of the execution time.

5. CONCLUSIONS

In this work, we designed a low power asynchronous IDCT circuit on an FPGA. The circuit was synthesized by using the maximum delay constraints for data-paths related to setup constraints. In the experiment, we confirmed that 24 % of power was reduced compared to the synchronous counterpart. However, the execution time and the energy consumption were increased due to the insertion of more LCELLs to delay elements.

As future work, we will optimize the execution time by assigning the maximum delay constraints for not only data-paths but also control paths.

6. REFERENCES

[1] J. Zhang, P. Chow, H. Liu, "FPGA Implementation of Low-Power and High-PSNR DCT/IDCT Architecture based on Adaptive Recoding CORDIC," Proc. IEEE International Conference on Field Programmable Technology 2015, pp. 128-135, Dec. 2015.

[2] T. Xanthopoulos, A. Chandrakasan, "A Low-Power IDCT Macrocell for MPEG-2 MP@ML Exploiting Data Distribution Properties for Minimal Activity," Journal of Solid State Circuits, vol. 34, pp. 693-703, May 1999.

[3] S. R. Bhaisare, A. V. Gokhale, P. K. Dakhole, "CORDIC Architecture Based 2-D DCT and IDCT for Image Compression," Proc. IEEE International Conference on Communication and Signal Processing 2015, pp. 1473-1477, Nov. 2015.

[4] G. K. Wallace, "The JPEG Still Picture Compression Standard," IEEE Transactions on Consumer Electronics, vol. 38, no. 1, pp. xviii-xxxiv, Feb. 1992.

[5] D. L. Gall, "MPEG: a video compression standard for multimedia applications," Communications of the ACM - Special issue on digital multimedia systems, vol. 34, no. 4, pp. 46-58, Apr. 1991.

[6] A. Rettberg, B. Kleinjohann, "A Fast Asynchronous Re-configurable Architecture for Multimedia Applications," Proc. 14th Symposium on Integrated Circuits and Systems Design 2001, pp. 150-155, Sep. 2001.

[7] K. Takizawa, S. Hosaka, H. Saito, "A Design Support Tool Set for Asynchronous Circuits with Bundled-data Implementation on FPGAs," Proc. IEEE 24th International Conference on Field Programmable Logic and Applications (FPL), pp. 1-4, Sep. 2014.

[8] Altera Cyclone IV, [https://www.altera.com/products/fpga/cyclone-series/cyclone-iv/features.html].

FPGA based Design of a Low Power Asynchronous MIPS Processor

Jukiya Furushima
University of Aizu
Aizu-Wakamatsu 965-8580, Japan
m5201118@u-aizu.ac.jp

Hiroshi Saito
University of Aizu
Aizu-Wakamatsu 965-8580, Japan
hiroshis@u-aizu.ac.jp

ABSTRACT

In this paper, we design a low power asynchronous MIPS processor on a Field Programmable Gate Array (FPGA). We synthesize the MIPS processor by assigning the maximum delay constraints for control paths and data paths. In the experiment, we evaluate the area, execution time, power consumption, and energy consumption of the designed MIPS processor comparing with the synchronous MIPS processor. We achieved 35% power reduction by the designed MIPS processor.

Categories and Subject Descriptors

B.6.0 [Hardware]: LOGIC DESIGN—General

General Terms

Design

Keywords

Asynchronous circuits, FPGA, processor design

1. INTRODUCTION

Most of currently used processors are synchronous circuits. Circuit components in synchronous circuits are controlled by global clock signals. In synchronous circuits, clock skew, power consumption, and electromagnetic radiation will be significant problems when semiconductor submicron technology is advanced more and more.

In asynchronous circuits, circuit components are controlled by local handshake signals. Asynchronous circuits are potentially low power consumption and low electromagnetic radiation due to the absence of global clock signals. However, the design of asynchronous circuits is more difficult than the design of synchronous circuits.

Several asynchronous processors have been proposed. Handshake Solutions designed a clockless ARM996HS using TiDE design flow [1]. Zhang and Theodoropoulos designed SAMIPS [2] using Balsa [3]. SAMIPS is an asynchronous MIPS processor similar to this paper. Amde et.al, designed an asynchronous DLX processor using Pipefitter [4]. Chang-Jiu et.al, designed an asynchronous 8051 microcontroller using Balsa [5]. Iwasaki designed an asynchronous AVR processor considering a cycle time constraint [6]. The targets of these researches were not FPGA but Application Specific Integrated Circuits (ASICs).

In this paper, we design a low power asynchronous MIPS processor on an FPGA. We synthesize the MIPS processor assigning the maximum delay constraints for data-paths and control paths. In addition, we evaluate the area, execute time, power consumption and energy consumption of the designed MIPS processor.

The organization of this paper is as follows. In section 2, we describe background used in this paper. In section 3, we describe the design of an asynchronous MIPS processor. In section 4, we describe the experimental result. In section 5, we conclude this work.

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

2. BACKGROUND

2.1 Asynchronous Circuits with Bundled-data Implementation

Figure 1 represents an asynchronous processor model with bundled-data implementation. The left side is the control circuit and the right side is the data-path circuit. The data-path circuit consists of Program Counter (PC), Memories (IMEM and DMEM), Instruction Register (IR), Decoder, Register File (RF), ALU, and delay elements hd_k. PC stores the address of the instruction memory. IMEM is a memory to store instructions. DMEM is a memory to store data. IR is a register to store an instruction fetched from IMEM. RF is a collection of registers. Data from DMEM and ALU are written into RF. hd_k are delay elements for registers or memories to guarantee hold constraints of registers. The control circuit consists of control modules $ctrl_i$ ($0 \leqq i \leqq m-1$). A control module $ctrl_i$ consists of a Q-module q_i, delay elements sd_i, bd_i, cd_i, and glue logics. sd_i is used to guarantee setup constraints. bd_i is used to guarantee the timing of a control branch. cd_i is used to guarantee the timing of resetting of control signals. Glue logics are branch decision logics and C-elements [8]. C-elements are synchronization components. The output of C-elements is 0 when all inputs are 0. The output is 1 when all inputs are 1. Otherwise, the output does not change. In the case of pipeline execution, two control modules $ctrl_{i\_1}$ and $ctrl_{i\_2}$ are used to control one pipeline stage.

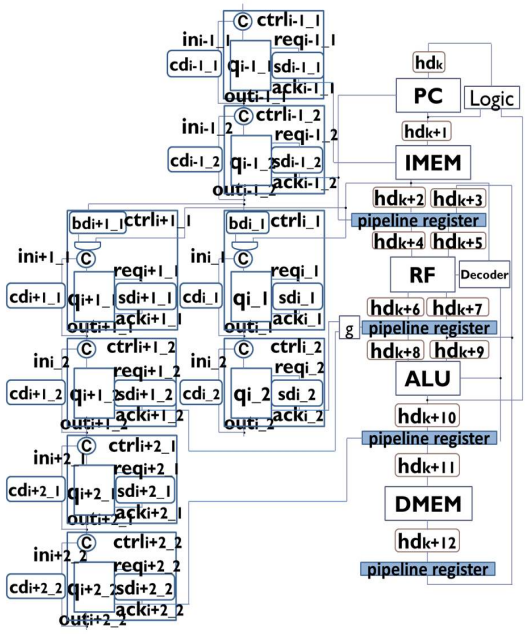


Figure 1: An asynchronous processor model with bundled-data implementation.

We describe behaviors of this processor model. Q-module q_i in $ctrl_i$ starts the control of operations in the data-path circuit when a rising edge in_i comes from the previous Q-module q_{i-1}. After the rising edge in_i signal arrives, q_i rises the request signal req_i, the signal passes sd_i, and rises the acknowledge signal ack_i, and go back to q_i. Then, q_i falls req_i signal and wait a falling edge of ack_i signal. Data are written into registers with the falling edge of ack_i. After ack_i returns q_i, q_i rises out_i and control signal moves to the next control module $ctrl_{i+1}$. In addition, out_i is returned to C-element to reset in_i and out_i.

The model needs to satisfy 5 types of timing constraints [6]. If some of them are violated, we need to adjust delay elements sd_i, hd_k, bd_i, cd_i to satisfy timing constraints.

2.2 The MIPS Processor

The MIPS processor used in this work consists of instruction fetch stage (IF), instruction decode stage (ID), execute stage (EX), memory access (MEM), and write back stage (WB) [9]. The block diagram of the MIPS processor is shown in Figure 2.

IF stage includes an instruction memory (IMEM), program counter (PC), an adder, and two multiplexers to decide the address of IMEM. ID stage includes a decoder, a register file (RF), a sign extension, a shifter, an adder, and a comparator. The decoder generates control signals from a fetched instruction. The register file saves data from a data memory (DMEM) or an arithmetic and logic unit (ALU). EX stage includes an ALU to execute arithmetic or logical operations. MEM stage includes a DMEM. In WB stage, data from ALU or DMEM are written into RF.

The MIPS processor supports 9 instructions. R type instructions (add, sub, or, and, slt) use register values. J instruction is a jump instruction. Beq instruction is a conditional branching instruction. Lw instruction reads data from DMEM and stores to a register. Sw instruction writes data in a register to DMEM.

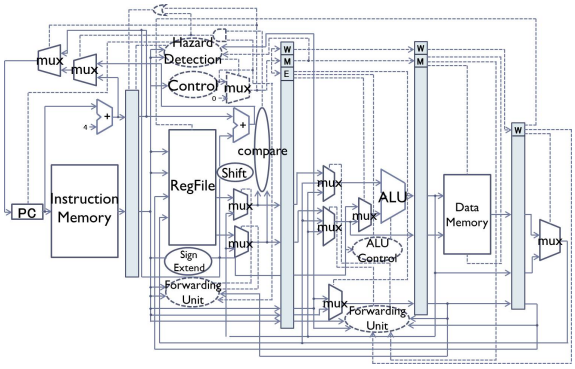


Figure 2: The block diagram of pipelined MIPS processor [9].

2.3 Field Programmable Gate Array

Field Programmable Gate Array (FPGA) is one of reconfigurable devices. FPGA has been used in many embedded systems because of the advantage such as lower design cost and flexibility to change circuit structure. Figure 3 shows the Altera Cyclone IV FPGA. The FPGA consists of Logic Arrays, Embedded Multipliers, Random Access Memories (RAMs), Input/output Elements (IOEs), and Phase Locked Loops (PLLs).

3. DESIGN OF AN ASYNCHRONOUS MIPS PROCESSOR

3.1 Design of an Asynchronous MIPS Processor Model

We design an asynchronous MIPS processor as follows. The data-path circuit is the same as the one used in [9]. The control circuit is modeled by a finite stage machine (FSM). Figure 4 shows an FSM of the MIPS processor. Nodes of the FSM represent pipeline stages. After IF stage, FSM is branched by a fetched instruction. Edges of the FSM represent dependencies between stages.

Figure 5 shows the generation of an asynchronous control circuit. We use the method in [6]. First, each node of FSM is divided into two. This is to hidden the initialization phase of control modules (i.e., reset of in_i and out_i). After division, a control module $ctrl_i$ is allocated to each node. If registers and memories are controlled by some control modules, write signals for them are generated by ack_i signals of the control modules. Finally, C-elements and feedback loops are inserted to control modules. Figure 6 shows the control circuit of the asynchronous MIPS processor.

Figure 7 shows the structure of $ctrl_i$ for Altera Cyclone IV FPGA. Primitives DLATCH and LCELL of Altera FPGAs are used. There are two DLATCHes in two C-elements. If there is no DLATCH in C-elements, timing analysis tool cannot analyze path delays due to the existence of a combinational loop. In addition, to perform timing analysis correctly, we set "synthesis_keep" attribute to the output of sd_i. Moreover, we insert two DUMMY modules to indicate a wire as a through point for path delay analysis. Delay

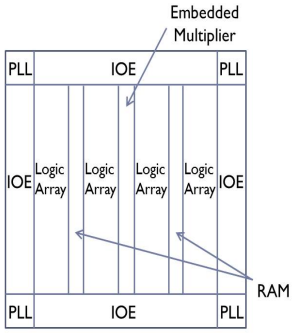


Figure 3: Structure of Altera Cyclone IV FPGA [10].

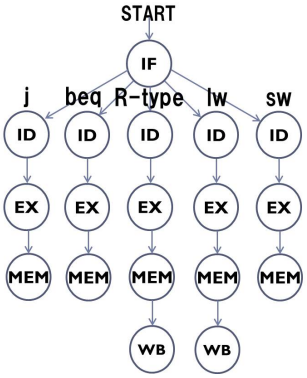


Figure 4: The FSM to represent pipeline stages of instructions.

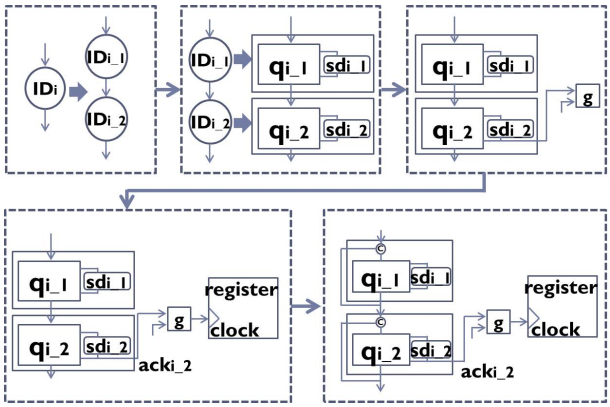


Figure 5: Generation of an asynchronous control circuit [6].

elements sd_i, hd_k, cd_i consists of LCELLs. They work as buffers. To avoid optimization for control modules, we set "Design Partition" for control modules.

Finally, we model the asynchronous MIPS processor using Verilog HDL.

3.2 Generation of the Maximum Delay Constraints and Setting of Margins

3.2.1 *Generation of the Maximum Delay Constraints*

To satisfy a given cycle time constraint CT, we assign the maximum delay constraints to paths related to setup constraints. Figure 8 represents a data-path $sdp_{i,p}$ and a control path $scp_{i,p}$ for a setup constraint. The data-path $sdp_{i,p}$ is divided into 2 sub-paths. The former is from sd_{i-1_2} to source register reg_{src} (PC in Fig. 8) and the latter is from reg_{src} to destination register reg_{dst} (IF/ID pipeline register in Fig. 8). The control path $scp_{i,p}$ is divided into 9 sub-paths: from sd_{i-1_2} to C-element in $ctrl_{i\_1}$, from C-element in $ctrl_{i\_1}$ to sd_{i_1}, from sd_{i_1} to C-element in q_{i_1}, from C-element in q_{i_1} to sd_{i_1}, from sd_{i_1} to C-element in $ctrl_{i\_2}$, from C-element in $ctrl_{i\_2}$ to sd_{i_2}, from sd_{i_2} to C-element in q_{i_2}, from C-element in q_{i_2} to sd_{i_2}, and from sd_{i_2} to reg_{dst}. The reason why we divide paths is to analyze path delays using general static timing analyzers correctly.

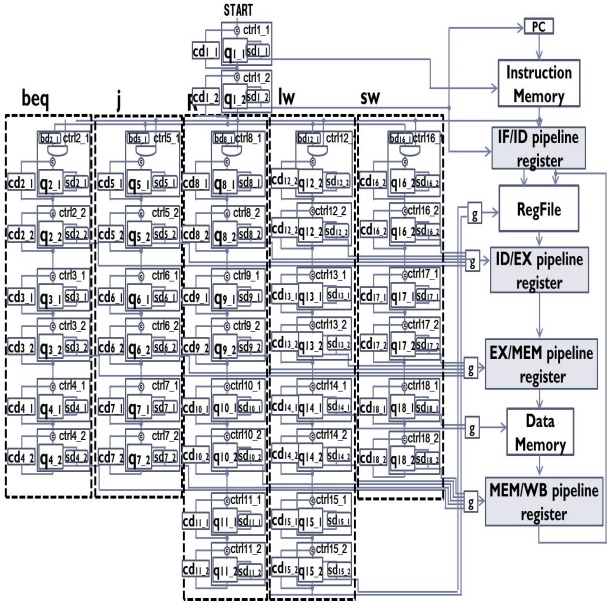


Figure 6: The control circuit of the asynchronous MIPS processor.

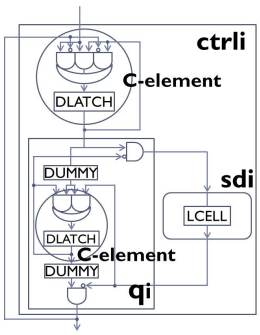


Figure 7: Structure of $ctrl_i$.

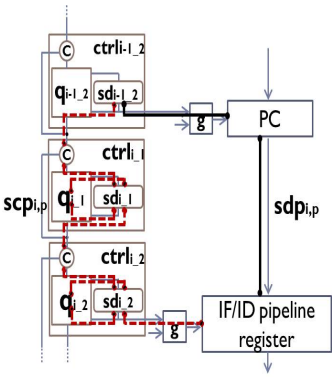


Figure 8: Paths to assign the maximum delay constraints.

We use the calculation method described in [11] to decide the values of the maximum delay constraints. In addition, we use *set_max_delay* command to represent the maximum delay constraint. *set_max_delay* command is provided in Synopsys Design Constraint (SDC) file format.

3.2.2 *Setting of Margins*

We setup margins for data-path delays and control path delays. For the former case, after FPGA implementation, there may be uncertain effects to data-path delays. We decide the margin based on a given cycle time constraint CT and actual path delays on an FPGA. We start from a small value. If there are timing violations during simulation, we increase the margin to solve timing violations.

In bundled-data implementation, from setup constraints, the minimum delay of $scp_{i,p}$ must be larger than the maximum delay of $sdp_{i,p}$ to write data into registers correctly. However, if the minimum delay of $scp_{i,p}$ is large enough for

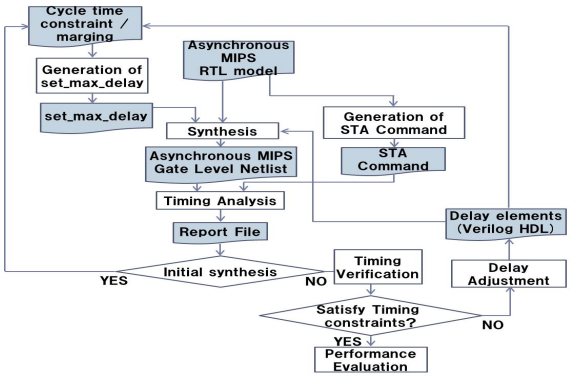


Figure 9: Design flow.

the maximum delay of $sdp_{i,p}$, it may results in performance degradation. Therefore, for the latter case, we decide the margin for control path delays from the delay of LCELLs and the number of delay adjustments to satisfy all timing constraints. We also start from a small value. If the number of delay adjustments for timing violations reported by static timing analysis overs a threshold value (e.g., 10 times), we increase the margin to satisfy timing constraints.

3.3 Design Flow

In this paper, we design an asynchronous MIPS processor based on a design flow shown in Figure 9. The inputs of the flow are a Verilog HDL model of the asynchronous MIPS processor explained in section 3.1, a cycle time constraint CT, and margins for data-path delays and control path delays.

The design flow is classified into the initial synthesis and the incremental synthesis. In the initial synthesis, we generate static timing analysis (STA) commands to analyze path delays related to timing constraints. Then, the initial synthesis and STA are carried out using Altera Quartus II and TimeQuest. From the STA results and cycle time constraint CT, we generate the maximum delay constraints using "set_max_delay" commands as described in section 3.2.

In the incremental synthesis, we repeatedly carry out synthesis and STA using the generated maximum delay constraints. If all of timing constraints are satisfied, we finish design. Otherwise, we carry out delay adjustment to satisfy timing constraints. If the number of delay adjustment overs a threshold value, we increase the margin for the control path delay to meet timing closure.

4. EXPERIMENT

In the experiment, we evaluate the designed asynchronous MIPS processor in terms of area, execution time, dynamic power consumption, and energy consumption comparing with a synchronous counterpart. The used synthesis tool and simulation tool are Altera Quartus II ver.14.1 and ModelSim-Altera ver.10.3.c. TimeQuest timing analyzer in Quartus II is also used to analyze path delays. The target device is Altera Cyclone IV (EP4CE115F29C7).

Initially, we synthesize the synchronous MIPS processor. We explore the best one in terms of clock frequency by

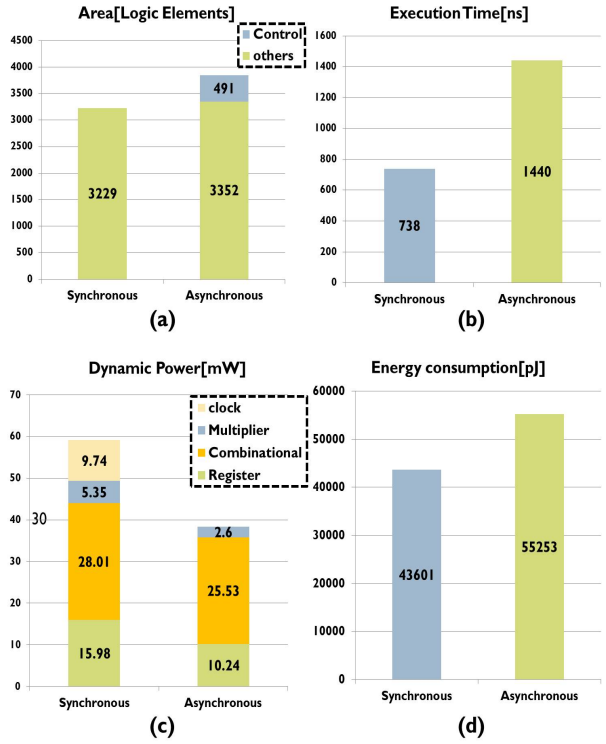


Figure 10: Experimental result: (a) area, (b) execution time, (c) dynamic power consumption, and (d) energy consumption.

changing clock constraint. The cycle time of the best one is 12 ns. We set 12 ns to CT and synthesize the asynchronous MIPS processor. The margin for data-path delays is set to 1.2 ns (10% of CT) and the margin for control path delays is set to 3.6 ns (30% of CT).

Figure 10.(a) shows the area in terms of the number of logic elements reported by Quartus II. The asynchronous MIPS processor requires 19% overhead. This mainly comes from the area of control modules and delay elements. Figure 10.(b) shows the execution time when a multiplication is given as a test input and simulated by using ModelSim-Altera. The asynchronous MIPS processor requires double of the synchronous counterpart. The increase of the execution time in the asynchronous MIPS processor comes from the margin for the control path delay. In fact, we set the margin to 4.8ns. The performance of the asynchronous MIPS processor will be the same as that of the synchronous MIPS processor if we synthesize the asynchronous MIPS processor with the margin of 2.4ns. This is our future work. Figure 10.(c) shows the dynamic power consumption obtained by PowerPlay Power Analyzer inside Quartus II. As we expect, 35% of power is reduced from the synchronous MIPS processor. This is mainly comes from the reduction of power consumption for the clock network and register. Finally, figure 10.(d) shows the energy consumption obtained by the product of execution time and dynamic power consumption. The energy is increased 27% due to the increase of the execution time.

5. CONCLUSIONS

In this paper, we designed a low power asynchronous MIPS processor on an FPGA. The asynchronous MIPS processor was synthesized with the maximum delay constraints for paths related to setup constraints. In the experiment, 35% dynamic power was reduced compared to the synchronous counterpart. However, the execution time and energy consumption were increased due to the insertion of many LCELLs to delay elements.

As future work, we improve the execution time of the designed MIPS processor by restricting the margin for the control path delay. In addition, we are going to reduce power consumption of the forwarding unit by removing unnecessary behaviors.

6. REFERENCES

[1] A. Bink and R. York, "ARM996HS: The First Licensable, Clockless 32-Bit Processor Core", Proc. Micro, pp. 58-68, Mar-Apr. 2007.

[2] Q. Zhang and G. Theodoropoulos, "MODELLING SAMIPS: A Synthesisable Asynchronous MIPS Processor", Proc. 37th Annual Simulation Symposium, pp. 205-212, 2004.

[3] D.Edwards and A. Bardsley, "Balsa : An asynchronous hardware synthesis language", The Computer Journal, 45(1): pp. 12.-18, 2002.

[4] M. Amde, I. Blunno and P. Sotiriou, "Automating the Design of an Asynchronous DLX Microprocessor", Proc. Design Automation Conference, pp. 502-507, 2003.

[5] Chang-Jiu Chen, Wei-Min Cheng, Ruei-Fu Tsai, Hung-Yue Tsai, and Tuan-Chieh Wang, "A Pipelined Asynchronous 8051 Soft-core Implemented with Balsa", Proc. APCCAS, pp. 976-979, 2008.

[6] S. Iwasaki, "Design and Evaluation of a Low Power Asynchronous AVR Processor considering a Cycle Time Constraint", Master Thesis, the University of Aizu, 2014.

[7] F. U. Rosenberger, C. E. Molnar, T. J. Chaney, and T. P. Fang, "Q-Modules:Internally Clocked Delay Insensitive Modules", IEEE Transaction of Computer, vol. C-37, no.9, pp. 1005-1018, 1988.

[8] D. E. Muller, and W. S. Bartky, "A theory of asynchronous circuits", Proc. International Symposium on the Theory of Switching, pp. 204-243, Apr 1959.

[9] D. A. Patterson and J. L. Hennessy, "Computer Organization and Design, Fifth Edition: The Hardware/Software Interface", Morgan Kaufmann, 2006.

[10] Altera Cyclone IV FPGA, "https://www.altera.com/products/fpga/cyclone-series/cyclone-iv/features.html"

[11] K. Takizawa, et al., "A Design Support Tool Set for Asynchronous Circuits with Bundled-data Implementation on FPGAs", Proc. IEEE 24th International Conference on Field Programmable Logic and Applications (FPL), pp.1-4, September 2014.

Using Ant Colony Optimization for Tourist Route Construction Automation

Boris Skripal
Peter the Great St. Petersburg Polytechnic University
29 Polytechnicheskaya st.
195251 St. Petersburg Russia
skipalboris@gmail.com

Evgeny Pyshkin
University of Aizu
Software Engineering Lab
Tsuruga, Ikki-machi, Aizu-Wakamatsu
Fukushima, Japan 965-8580
pyshe@u-aizu.ac.jp

ABSTRACT

In this paper an ant colony optimization (ACO) adoption for tourist itinerary planning automation is presented. We introduce a model describing a set of features used in our component for itinerary planning automation and study how an ACO algorithm could be used for the purposes of constructing an attractive tourist route.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Automation, Human Factors

Keywords

Information systems, Travel itinerary, ACO (Ant Colony Optimization)

1. INTRODUCTION

Tourist route planning automation is often considered as a generalization of shortest path algorithms, however there are many competitive factors making difficult to rely only on the minimization algorithms straightforwardly. Many connected aspects have to be taken into consideration including but not limited to the following ones:

- Tourists usually don't want to reach a point of interest in a shortest possible time, they rather expect to arrange their leisure time in the best way.
- Departure and arrival points are often the same (so, you would probably like to be back to your hotel after discovering a certain set of attractions).
- Besides walk distance, there are other factors affecting an itinerary score: object interestingness, ticket prices and related expenses, probability of being too crowded, opening hours, etc.
- A route shouldn't include only historical or cultural attractions, food, relaxation and sanitary points are also important.
- A route might be strongly focused (in a case when it includes a selection of objects of similar type, e.g. mostly churches, or monuments, or art exhibitions, etc.) or less focused (if there is a balance between attractions of different type).

More aspects we take into consideration, more accurate planning we can expect. There are at least three major problems to be resolved while implementing route planning systems including a problem of mining data required for a future route [5], a problem of constructing an algorithm for itinerary planning, and a problem of implementing a good presentation of the proposed route [4].

In this paper we pay a particular attention to developing an algorithm for planning tourist routes on the base of an ant colony optimization modification aimed at taking into accounts some competitive factors listed above.

Based on [2] we use a model of a tourist route [1] represented as a hierarchy of concepts describing tourist-related information (such as expenses, speed and energy), attraction-related information (such as name, type, visiting fees, walking time, location and attractiveness score) and visit-related information (such as a list of included attractions, visit time and duration).

2. RELATED WORK

The standard task of tourist route generation is formalized by Souffriau [7] as follows:

Assume there are N points of interest (POIs), each point can be connected to other ones (but a departure point isn't connected with a destination point). Each POI i has a score $S_i \geqslant 0$, where for a departure point $i = 1$, for the destination point $i = N$. The shortest path between points i to j requires time t_{ij}, the total score S_{total} has to be maximized under a time limit constraint T_{max}. This model sets the future route boundaries and determines criteria for the best tourist route such as a best attraction selection-best path combination.

[1]The proposed model is not complete, but provides a set of basic features which are enough to illustrate our vision.

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

In this study we investigate a possibility to adopt an ACO algorithm for automated itinerary construction. A basic ACO algorithm (usually used for finding shortest paths in graphs [8]) is based on a probability of ant moves from state x to state y:

$$p_{xy}^k = \frac{(\tau_{xy}^\alpha)(\eta_{xy}^\beta)}{\sum_{z \in allowed_z}(\tau_{xz}^\alpha)(\eta_{xz}^\beta)} \quad (1)$$

In equation (1) α and β are coefficients used to manage τ_{xy} and η_{xy}, τ_{xy} representing the amount of pheromone deposited for transition from state x to y and η_{xy} – the desirability for xy transition.

A pheromone update conditioned by a transition is as follows:

$$\tau_{xy}(t+1) = (1-p)\tau_{xy}(t) + \sum_k \Delta\tau_{xy}^k(t) \quad (2)$$

In equation (2) $0 < p < 1$ is the pheromone trail evaporation, $\Delta tau_{xy}^k(t)$ is the amount of pheromone that an ant k puts on the arcs it visits (defined in the formula 3).

$$\Delta\tau_{xy}^k(t) = \begin{cases} 1/L^k(t), & \text{if the path } \{xy\} \text{ is used by the ant } k \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

Due to the randomization and a kind of feedback based schema, ACO allows discovering good solutions rapidly without loosing local maximums. This is a reason to use ACO as a foundation for our purposes.

One known modification of ACO for creating tourist routes was described by Huang Han-Chen [3]. Its general idea is to extend a pheromone update function 2 by an additional parameter $\Delta\tau_{xy}^* = \sigma\frac{Q}{L^*}$, where σ is the number of elite ants, L^* is the route length of the determined optimal solution.

Another modification was introduced in [1] by Claes and Holvoet. Their idea is to extend the list of basic ACO parameters by adding transition speed from i-th vertex to j-th, as well as the path time and some others.

3. ADOPTING ACO FOR ROUTE CONSTRUCTION AUTOMATION

In order to pay attention to the POI types, we defined a table of preferences containing the preference coefficients for different types of objects (see Table 1 as an example).

Table 1: Preference Coefficients

Museums	Monuments	Parks	Restaurants	Cafes
1	0.8	1.2	0.6	0

There is a multiplication coefficient $\acute{p}$: $0 \leq \acute{p}$ for each type of objects, so as 0 corresponds to the lowest preference level (such an object will never be selected).

In order to improve algorithm performance we make preliminary selection of the objects considered for inclusion to the route. As we described in [6], for object selection we propose to use an elliptical area so as to allow selecting reachable objects which are inside the ellipse. The reason of using an elliptical area is that an ellipse has such a nice property that the sum of distances from every point on the curve to two focal points is constant. Hence, the sum of distances to the two focal points is set as a maximum possible distance: any point inside the curve can be reached while points outside the curve are unreachable. Semi-major axis (a) and semi-minor axis (b) are defined as follows:

$$a = \sqrt{(X_{start} - X_{finish})^2 + (Y_{start} - Y_{finish})^2} \quad (4)$$

$$b = \frac{Vt}{2} - \frac{a}{2} \quad (5)$$

In equation (5) V is the tourist speed, t is the planned route duration.

For a point P_i represented by its coordinates x and y, "being inside the ellipse" condition is defined as follows:

$$\frac{(\acute{x}\cos\alpha + \acute{y}\sin\alpha)^2}{a^2} - \frac{(-\acute{x}\sin\alpha + \acute{y}\cos\alpha)^2}{b^2} \leq 1 \quad (6)$$

In equation (6) $\acute{x} = x - x_0$, $\acute{y} = y - y_0$ (where x_0 and y_0 are the coordinates of the ellipse center).

The points which are outside the elliptic area are automatically excluded from the searching process.

In addition to standard ACO parameters (see equation (1)), each ant is associated with an energy reserve value E which allows taking into account such human (not ant!) properties as being hungry or/and being tired.

Effectively, in i-th step, there is a probability e_i that the next selected point is a food place:

$$e_i = \frac{E_{init} - E}{E_{init}} \quad (7)$$

In equation (7) E_{init} is an initial energy reserve, E is a current energy level. Each time unit (for example, each one minute of walking) the energy reserve score decreases by some value (now we use a fixed value, but it could be also a function of a distance). After visiting a food place, the energy level is restored to E_{init}.

Total score of each route is calculated by using the following equation:

$$S = \frac{t_{see}}{t_{moving}}(-f_e t_{ze} + \sum_{i \in visited} S_i\acute{p}) \quad (8)$$

As you see from equation (8), if E becomes zero, we decrease ant score S_{total} by f_e (f_e being a penalty per time unit).

To calculate the total score, the pheromone level on the best routes is updated after each iteration.

4. A PROCESS OF ROUTE GENERATION

There are some preliminary actions before generating a route. First, we prepare a table of preference (see Table 1). Second, some parameters have to be set:

- Route speed (we use 5 kph as a default value for average pedestrian speed);
- Excursion departure and destination points (they may be the same);
- Excursion departure time and expected arrival time to the destination [2];

[2]A set of possible destinations could also be an option. For example, in a big city a traveler might prefer to reach a metro station: sometimes it doesn't matter which one.

- List of filtered POIs (due to being closed or not fitting well user preferences);
- Tourist balance (a non-negative value representing how much money a traveler plans to spend for an excursion);
- Initial energy value and an energy unit used while decreasing the energy level.

Figure 1 illustrates the process of selecting points with the scope of elliptic area and with paying attention to preference constraints.

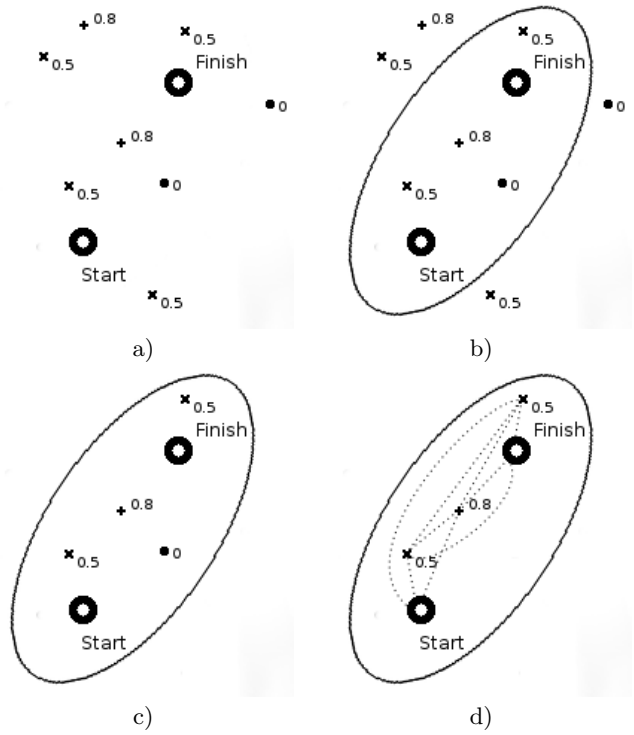


Figure 1: Selecting POIs: a) initial set of POIs (three POI types shown), b) reachable area, c) filtering unreachable POIs, d) removing POIs with zero preference level or those which aren't open for visiting (dashed lines show possible ways between points).

We start a route generation procedure from creating some ants and setting required parameters (initial time, departure point, etc.). For each ant there are the following steps:

1. If there are POIs which are unreachable within the limits of the planned period of time (for example, they might be closed or they might delay significantly the total route duration due to a long waiting time at the entrance) such POIs are filtered.
2. If there is no more POIs to consider, move to the finish and stop searching.
3. If there is an option to include a food point (and there are some available), select a food point with probability e_i calculated by using equation (7) (otherwise, an attraction point is selected, respectively, with probability $1 - e_i$.
4. Proceed with a standard ACO iteration: select the next POI to visit by using equation (1) and move to the selected POI.
5. Update current ant time ($t_{new} = t_{prev} + t_{moving} + t_{visit}$), the energy level, as well as moving time and visiting time (if we visit a food place, we don't include object visiting time), balance and pheromone on this point by using formula (2).
6. Go to the step 1.

When all ants achieved the finish point, we select N best routes (each route is evaluated by using formula (8)) and move "elite" ants on these routes, for update best routes (increase τ of all POIs in the best path).

Then a next set of ants is generated and the above procedure is repeated. The final route is selected among the best routes obtained during the above described iterations.

5. CONCLUSIONS

In this paper we described an ACO modification to be used for tourist route generation. Further investigations are required in order to evaluate whether such an approach is appropriate for creating interesting routes under a selection of constraints and route properties. In the future we plan to consider such parameters as season dependency, visiting time periods (to avoid crowds), leveraging information about tourist preferences achieved from thematic web sites. An interesting issue is generating multi-day or thematically connected routes considered as the parts of one journey.

6. REFERENCES

[1] R. Claes and T. Holvoet. Ant colony optimization applied to route planning using link travel time predictions. *IEEE International Parallel and Distributed Processing Symposium*, 2011.

[2] D. Gavalas, M. Kenteris, C. Konstantopoulos, and G. Pantziou. An efficient algorithm for recommending personalized mobile tourist routes. *IET Software*, 2011.

[3] H. Huang. The application of ant colony optimisation algorithm in tour route planning. *Journal of Theoretical and Applied Information Technology*, 2013.

[4] Y. Huang, C. Liu, C. Lee, and Y. Huang. Designing a personalized guide recommendation system to mitigate information overload in museum learning. *Educational Technology and Society, 15*, 2012.

[5] M. Karathiya. Data mining for travel and tourism. *Journal of Information and Operations Management*, 2012.

[6] B. Skripal and E. Pyshkin. Automated leisure walk route generation for an interactive travel planner. In *Proceedings of the International Workshop on Applications in Information Technology (IWAIT-2015)*, pages 29–32. The University of Aizu, The University of Aizu Press, Oct 2015.

[7] W. Souffriau. *Automated Tourist Decision Support.* Katholieke Universiteit Leuven, 2010.

[8] T. Stutzle and M. Dorigo. Ant colony optimization: Overview and recent advances. *IRIDIA, Institut de Recherches Interdisciplinaires et de Developpements en Intelligence Artificielle*, 2009.

Rating Prediction Operation of Multi-criteria Recommender Systems Based on Feedforward Network

Mohammed Hassan
Software Engineering Lab
University of Aizu
Aizu-Wakamatsu, Japan
d8171104@u-aizu.ac.jp

Mohamed Hamada
Software Engineering Lab
University of Aizu
Aizu-Wakamatsu, Japan
hamada@u-aizu.ac.jp

ABSTRACT

Recommender systems are software systems that have been widely used to recommend items to the user. They have the capacity to support and enhance the quality of decisions people make when finding selecting items online. The most common techniques used by many recommendation systems are collaborative filtering, content-based, knowledge-based and hybrid-based which combines two or more techniques to make predictions and recommendations.

Multi-criteria recommendation technique is a new technique used to recommend items to users based on multiple attributes of items. This technique has been used and proven by researchers in industries and academic institutions to provide more accurate predictions and recommendations than the traditional techniques. What is still not yet clear is the role of some machine learning algorithms such as artificial neural network to improve its prediction accuracy. This paper proposed using feedforward neural network to model user preferences in multi-criteria recommender systems. The operational results of experiments for training and testing the network using two training algorithms and yahoo movie data sets are also presented.

Categories and Subject Descriptors

H.4.2 [**Information Systems and Applications**]: Decision Support; I.2.1 [**Artificial Intelligence**]: Application and Expert Systems

General Terms

Recommender systems, Algorithms, Experimentation

Keywords

Recommender systems, Artificial neural network, Prediction Accuracy

1. INTRODUCTION

A recommender system is an intelligent system that plays an important role in providing suggestions of valuable items to users. Such suggestions take the forms of different processes of decision making, like the kind of movie to watch, a music to listen, items to buy, or an online news to read [17, 12, 16]. Recommender systems are classified based on the technique used to design the system. Traditionally, collaborative filtering, content-based, knowledge-based, and hybrid-based are the major techniques used to describe the name or nature of the system. Therefore, knowing the recommendation techniques is at the heart of our understanding of recommender systems. Those techniques are sometimes called traditional techniques, and are increasingly becoming popular ways of building a system that combat problems of information overload [7, 5].

However, despite their popularity and providing considerable prediction and recommendation accuracies, they suffer from major drawbacks [1, 4, 13] because they work with just a single rating, whereas most of the time the acceptability of the item recommended may depend on several item's attributes [3]. Researchers suggested that if ratings provided to those several characteristics of items will be considered during predictions and recommendation process, it could help to enhance the quality of recommendations since complex opinions of users will be captured from various attributes of the item. Recent developments in this field have led to the existence of new recommendation technique known as multi-criteria recommendation technique [1, 3] that exploits multiple criteria ratings from various item's characteristics to make recommendations. This technique has been used for wide range of recommendation applications such as recommending products to customers [15, 11], hotel recommendation for travel and tourism [8], and so on.

Having considered multi-criteria technique as a resolution to some flaws of traditional techniques, it is also logical to look at various ways of modeling the multiple ratings to enhance the prediction accuracies and recommendation qualities. However, few researchers have been able to advance on some systematic research into improving the accuracy [9]. In addition, no previous research has investigated the effect of using artificial neural network to model user's preferences in order to improve the operations of multi-criteria recommender systems [3]. Therefore, in our quest to compensate for this knowledge gap, this study seeks to shine new light on the best way to use feedforward network through examination of the performance of backpropagation and delta rule algorithms to train the network using multi-criteria rating data set for recommending movies to users based on four

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

attributes of movies. This paper has been divided into four parts including this introduction section. The second part of the paper gives a brief literature review. The experimental methodologies are contained in the third part while the fourth section displays the result and discussions and the final section is concerned about conclusion and presenting future research work.

2. LITERATURE REVIEW

To be able to understand the concept of recommender systems, we introduced some mathematical notations μ, ι, δ, and ψ to represent the set of users, the set of items, a numerical rating, and a utility function respectively. The notation δ (the rating) is the measure of the degree to which a user in the set μ will like and item in ι, while the utility function ψ is a mapping from a $\mu \times \iota$ pair to a real number δ written as $\psi : \mu \times \iota \mapsto \delta$. The value of δ is a real number within a specifically defined interval such as between 1 to 5, 1 to 13, or it can be represented using non-numerical values such as like, don't like, . . ., strongly like, true or false, and so on [14]. Therefore, recommender systems try to predict the value δ of items in ι that have not been seen by the user and recommend those with a high value of δ.

The methods of prediction and recommendation explained in the above paragraph are the mechanisms followed essentially by traditional recommendation techniques. Moreover, a similar approach is followed by multi-criteria recommendation technique with the distinction that it uses multiple values of δ for each $\mu \times \iota$ pairs. In multi-criteria technique, the utility function ψ can be defined using one of the relations in equation 1.

$$\begin{aligned} &\psi : \mu \times \iota \mapsto \delta_0 \times \delta_1 \times \delta_2 \times ... \times \delta_n \\ &OR \\ &\psi : \mu \times \iota \mapsto \delta_1 \times \delta_2 \times \delta_3 \times ... \times \delta_n \end{aligned} \tag{1}$$

It is important to note however, that the two relations in equation 1 are different due to the presence of δ_0 in the first mapping. There are $n + 1$ and n ratings in the relations respectively. The additional rating δ_0 is called the overall rating which is either to be obtained from the user together with other n ratings or its value need to be computed based on the other n values as in equation 2. Detail explanation of multi-criteria recommendation is beyond the scope of this paper, readers can refer to [3, 2, 13] for more information.

$$\delta_0 = f(\delta_1, \delta_2, \delta_3, ..., \delta_n) \tag{2}$$

The technique can work even without taking δ_0 into account so that there is no overall rating, only ratings of other attributes will be used to undertake the operation process. However, evidence observed from many researchers confirmed the efficiency of considering the overall rating than ignoring it [1].

Nevertheless, neural network is one of the powerful classes of machine learning models that can learn a complicated function from a data to solve many optimization problems. A neural network aimed to mimic the functions of biological neurons [6]. It contains sets of connected neurons arranged in a layered style (Figure. 1), where an input layer consist of neurons that receive input from external environment and the output layer neuron receives the weighted sums of the products of input values and their corresponding weights from the previous layer and send its computational result to

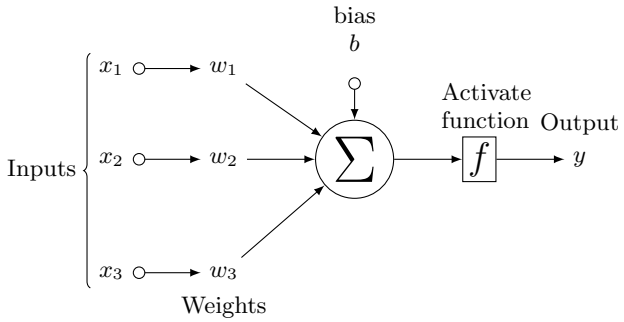


Figure 1: Simple Architecture of Feedforward Network

the outside environment.

The features x_1, x_2, and x_3 in Figure. 1 are inputs presented to the input layer, the parameters ω_1, ω_2, and ω_3 are the synaptic weights for links between input and output neurons. $\sum$ is the weighted sum of $\omega_i x_i$ for $0 \leqslant i \leqslant 3$ including the bias b and f is an activation function (non-linear differentiable function) that estimate the output y of the network. It could also be said that the output y can be written as $f(\sum_{i=0}^{n} \omega_i x_i)$. Feedforward network may contain more than two layers, where hidden layer(s) can be added between the input and output layers. Before the network gives the optimal output, it has to undergo a series of training using recommended algorithms. This is a brief abstract notion of how neural network behaves, details of this process can be found in [6, 19].

3. EXPERIMENT

The experiment was carried using yahoo movie datasets [11] for multi-criteria movie recommendation system, where movies are recommended to users based on four characteristics of movies, namely, action, story, direction, and visual effect of the movie which are represented as c_1, c_2, c_3, and c_4 respectively. In addition to those four criteria, an additional rating c_o called overall rating criterion was used to represent the final user's preference on a movie. The criteria values (ratings) in the dataset were initially presented using 13-fold quantitative scale from A+ to F representing the most and the worst preferences of the user. In the same manner, we changed the rating representation to numerical form (13 to 1 instead of A+ to F), and a total of approximately 63,000 ratings were used for the study. The target of the study was to use feedforward network to learn how to estimate c_o from c_1, c_2, c_3, and c_4. The dataset was divided into training and test data in the ratio of 75:25 for all the two experiments.

Two feedforward networks were developed using object oriented programming techniques in java [10] with learning capacities in delta rule and backpropagation. The Adaline network consists of input and output layer as in Figure 1, with input layer containing four neurons and a bias for passing the data to the output layer. A linear activation function f was used in the output neuron to process the weighted sum $(\sum_{i=1}^{5} x_i \omega_i)$ of the inputs x_i received from input layer. Furthermore, in addition to the two layers in the Adaline, a network containing additional hidden layer

with the same number of neurons as the input layers was used for backpropagation training with additional activation function g (sigmoid function) that receives the weighted sum from the input layer and send the result of its computation to the output neuron. For measuring the training and test error, a mean square error ($MSE = \frac{1}{2N}\sum_{j=1}^{N}(y_j - o_j)^2$) for real output o_j and the estimated output y_j, were used to compute the errors. Pearson correlation coefficient ($Corr = \frac{\sum(y_j-\overline{y})(o_j-\overline{o})}{\sqrt{\sum(y_j-\overline{y})^2}\sqrt{\sum(o_j-\overline{o})^2}}$) were also used as a metric for measuring the relative relationship between the real and estimated output for the test data.

Finally, in order to make the training faster and to avoid the chances of getting stuck in local optima, the input data were normalized to real numbers within the interval (0, 1] instead of between 1 and 13 inclusive.

4. RESULT AND DISCUSSION

In each of the two algorithms, neurons weights ω_i were initially generated at random and the network compute the outputs and the corresponding errors (as $\frac{1}{2}(y_j - o_j)^2$). Iteratively, the algorithms search for a set of weights ω_i $i = 1, 2, ..., 5$ that minimize the error. The adaptive linear neuron (Adaline) network trained using delta rule shows a quick convergence within a few number of iterations (less than 10 iterations) with a very performance. On the other hand, backpropagation algorithm prolongs the learning process where a large number of training cycles (epochs) have been used to monitor its performance and the result is presented in Figure 2. This figure shows the average MSE for the various number of training cycles. It is apparent from the figure that the number of training cycles is inversely proportional to the errors observed. It shows that the convergence can only be attained at a very high number of iterations. However, for the purpose of comparison, the number of training cycle was set 10,000 cycles (epoch = 10,000), the training error and correlation between the real and estimated output of the test set for the two algorithms are shown in Table.1.

Table 1: **Performance Statistics**

Algorithm	Number of Iterations	Average Training MSE ($\times 10^{-3}$)	Percentage correctness
Adaline	10	5.34	94.4%
BPA	10,000	7.30	90.0%

5. CONCLUSION AND FUTURE WORK

This study was carried out to investigate the relative performance of single layer and multilayer feedforward network trained using delta rule and backpropagation algorithm respectively. The performance of each model was measured using MSE for the training and the percentage of the correct predictions were evaluated on the test data using Pearson correlation coefficient. From Figure.2 and Table.1, it can be seen that backpropagation algorithm has a greater demand for longer training cycles to converge. Moreover, the results indicate that the one layer network trained using adaptive linear neuron algorithm is more efficient than the two layer network which supports the traditional belief that single layer network produces less error than multilayered network [18]. Up to our last experiment with epochs of 10,000, backpropagation did not completely show the final convergence, further investigation is recommended to estimate the approximate epochs required by the algorithm to converge and to know whether it will produce a better result than Adaline. The study confirmed the usefulness of training a neural network model with features of inputs obtained for predicting the user preferences on items based several characteristics of items in multi-criteria recommender systems.

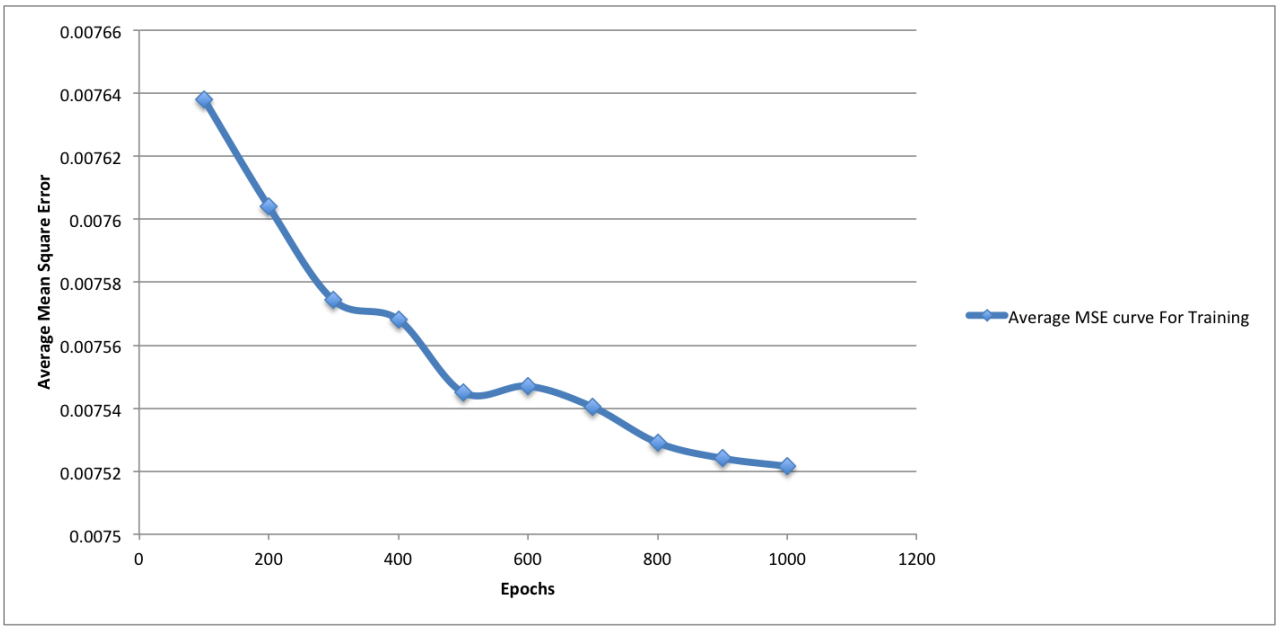


Figure 2: Average training MSE for Backpropagation

REFERENCES

[1] G. Adomavicius and Y. Kwon. New recommendation techniques for multicriteria rating systems. *IEEE Intelligent Systems*, 22(3):48–55, 2007.

[2] G. Adomavicius, N. Manouselis, and Y. Kwon. Multi-criteria recommender systems. In *Recommender systems handbook*, pages 769–803. Springer, 2011.

[3] G. Adomavicius, N. Manouselis, and Y. Kwon. Multi-criteria recommender systems. In *Recommender systems handbook*, pages 854–887. Springer, 2015.

[4] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*, 17(6):734–749, 2005.

[5] N. Farokhi, M. Vahid, M. Nilashi, and O. Ibrahim. A multi-criteria recommender system for tourism using fuzzy approach. *Journal of Soft Computing and Decision Support Systems*, 3(4):19–29, 2016.

[6] D. Graupe. *Principles of artificial neural networks*, volume 7. World Scientific, 2013.

[7] M. Hassan and M. Hamada. Recommending learning peers for collaborative learning through social network sites. *IEEE ISMS, Intelligent Systems, Modeling and Simulation*, 2016.

[8] D. Jannach, F. Gedikli, Z. Karakaya, and O. Juwig. *Recommending hotels based on multi-dimensional customer ratings*. na, 2012.

[9] D. Jannach, Z. Karakaya, and F. Gedikli. Accuracy improvements for multi-criteria recommender systems. In *Proceedings of the 13th ACM Conference on Electronic Commerce*, pages 674–689. ACM, 2012.

[10] S. Kendal. *Object Oriented Programming using Java*. Bookboon, 2009.

[11] K. Lakiotaki, N. F. Matsatsinis, and A. Tsoukias. Multicriteria user modeling in recommender systems. *IEEE Intelligent Systems*, 26(2):64–76, 2011.

[12] T. Mahmood and F. Ricci. Improving recommender systems with adaptive conversational strategies. In *Proceedings of the 20th ACM conference on Hypertext and hypermedia*, pages 73–82. ACM, 2009.

[13] N. Manouselis and C. Costopoulou. Analysis and classification of multi-criteria recommender systems. *World Wide Web*, 10(4):415–441, 2007.

[14] X. Ning, C. Desrosiers, and G. Karypis. A comprehensive survey of neighborhood-based recommendation methods. In *Recommender systems handbook*, pages 37–76. Springer, 2015.

[15] K. Palanivel and R. Sivakumar. A study on collaborative recommender system using fuzzy-multicriteria approaches. *International Journal of Business Information Systems*, 7(4):419–439, 2011.

[16] D. H. Park, H. K. Kim, I. Y. Choi, and J. K. Kim. A literature review and classification of recommender systems research. *Expert Systems with Applications*, 39(11):10059–10072, 2012.

[17] F. Ricci, L. Rokach, and B. Shapira. Recommender systems: introduction and challenges. In *Recommender Systems Handbook*, pages 1–34. Springer, 2015.

[18] A. M. Souza and F. M. Soares. *Neural network programming with Java*. Packt Publishing Ltd, 2016.

[19] P. D. Wasserman. *Advanced methods in neural computing*. John Wiley & Sons, Inc., 1993.

Music Emotion Recognition

Gao Feng
School of Computer Science and Engineering
University of Aizu
Aizu-Wakamatsu City
Fukushima-ken, Japan
m5191104@u-aizu.ac.jp

Konstantin Markov
School of Computer Science and Engineering
University of Aizu
Aizu-Wakamatsu City
Fukushima-ken, Japan
markov@u-aizu.ac.jp

Jianguo Yu
School of Computer Science and Engineering
University of Aizu
Aizu-Wakamatsu City
Fukushima-ken, Japan
m5182104@u-aizu.ac.jp

ABSTRACT

In this paper, we describe our approaches for the MediaEvals' 2015 "Emotion in Music" task. Emotion analysis and recognition have become an interesting issue of research in the middle of the computer vision research area. Our methods consist of Multivariate Linear Regression (MLR), Support Vector Regression (SVR) and FeedForward Neural Networks (FFNN) for dynamic Arousal and Valence regression. In this paper, we first present the results by using the MLR and SVR, then present the results of FFNN. The recognition of music emotions using Deep Learning is one of the latest challenges in the field of speech processing. The simulation results show that recognition with FFNN is better than the traditional methods (MLR and SVR).

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Pattern RecognitionDeep Neural Network; D.2.8 [**Software Engineering**]: Music—*emotion recognition, deep learning*

General Terms

Deep Neural Network

Keywords

Emotion recognition, arousal, valence, linear regression, SVR, FFNN

1. INTRODUCTION

Emotion is a term for a psychological and physiological state associated with a broad variety of thoughts, feelings, and behaviors a [1]. Emotions are subjective experiences, or experienced from an individual point of view. Emotion is often associated with mood, temperament and personality. But in general emotions are short-term whereas moods are long-term and temperaments or personalities are very long-term. Human emotion can be of different types such as happiness, angry, fear, sadness, surprise, disgust, bored, shy etc. Recently, music and emotion recognition, which tries to recognize emotion from music signals, has received increasing attention. Music emotion recognition is a very challenging task of which extracting effective emotional features is an open question [2][3].

Nowadays, there are huge amount of speech and music data on the internet. Thus, availability of automatic systems which can estimate human emotions from speech and music can play an important role in developing new sophisticated applications and services in entertainment as well as in health care industries. A very important problem of automatic speech and music emotion recognition is that the emotion is not only subjective, but also difficult to quantify and analyze. Therefore, the establishment of flexible emotional models is highly demanded. A deep neural network (DNN) is a feed-forward neural network that has more than one hidden layer between its inputs and outputs. With sufficient training data and appropriate training strategies, FFNNs perform very well in many machine learning tasks [4].

2. RELATED WORKS

In the past ten years, there has been a rapid expansion of music information retrieval research towards automated systems with the processing of vast and easily-accessible digital music libraries. Recognition of emotions in music is still in processing, though it has received increasing attention recently [5]. Determining the emotional content of music audio computationally is a cross disciplinary endeavor spanning signal processing, machine learning, music theory and auditory perception. Computational systems for music mood recognition may be on the basis of emotion model, which remain an active topic of psychology research. Categorical and parametric models are supported through substantial prior research with human subjects. Both models are used in Music-IR systems, but the collection of "ground truth" emotion labels remains a particularly challenging problem regardless of the representation being used. The annual Music Information Research Evaluation eXchange (MIREX) is a community-based framework for evaluating Music-IR systems [6]. It included audio music mood classification as a task for the first time in 2007 [7]. The highest performing systems in this category demonstrate the improvement using solely acoustic features every year. But the emotion is not completely encapsulated within the audio alone(social context, for example, plays a prominent role), so approaches incorporating music metadata, such as tags and lyrics.

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

3. METHDOLOGY

3.1 Deep Neural Networks

Deep Neural Networks (DNNs) have recently achieved breakthrough results in almost every machine-learning task. They are a set of machine learning algorithms inspired by how the brain works. Unlike most traditional machine-learning algorithms, Deep Neural networks perform automatic feature extraction without human interference. [8] A simple DNN shows in Fig.1

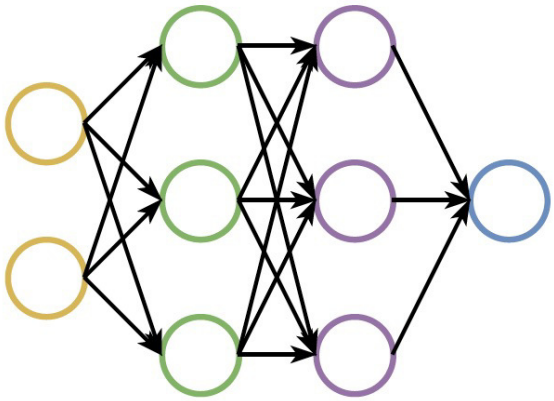


Figure 1: simple DNN

It is hard to understand the behavior of deep neural networks in general, but it is much easier to see what is happening when the data passed through a single layer: A mapping from the input space to the output space. It is a basally linear transformation followed by an activation function, whose mathematical description is Eq.(1), where $\vec{x}$ is the input, W is weights matrix, $\vec{b}$ is bias, a() is activation function and $\vec{y}$ is the output of this layer.

$$\vec{y} = a(W \times \vec{x} + \vec{b}) \tag{1}$$

The single layer actually transforms the data and create a new representation. The equation describe a process by the 5 space operations. First, change the dimensionality of the input space. Second, Rotate the input space. Third, Scale the input space. Fourth, Translate the input space. Fifth, "bend" the input space. The first 3 operations are done by W*$\vec{x}$, the 4th one is provided by $\vec{b}$, and the 5th operation is done by a() which gives nonlinearity to the layer.

3.2 Data Pre-Processing

Like other machine learning methods, pre-processing is needed before training.

Mean Subtraction

Mean subtraction is the most common way which can make it easy for the network to converge. It is just subtracting the mean across every individual feature in the data and it can center the cloud of data around the origin along every dimension.

Normalization

Normalization is the process of organizing the columns (attributes) and tables (relations) of a relational database to minimize data redundancy. There are two common ways of achieving this:

1. Divide each dimension by its standard deviation, once it has been a zero-centered.

2. Normalize each dimension so that the min and max along the dimension is -1 and 1.

Notes that the outputs should also correspond to the activation function of the output layer. For example, if we use the sigmoid activation function, then the range of outputs should be (0,1).

One-Hot Vector

If the task is classification, then instead outputting a most likely class, we also want to know the probabilities being other classes. Then, we need to convert our labels to one-hot vectors of size number of classes. For example, the class with index 12 would be the vector of all 0's and a 1 at position 12.

PCA and Whitening

PCA and Whitening are another form of preprocessing that also helps the convergence of FFNNs.

3.3 Activation functions

The mostly used activation functions are **ReLU, Sigmoid, Tanh**. ReLU has two benefits. First, it is fast than the other two. Second, it does not suffer from the vanishing gradient problem. ReLU's are faster to compute because 1) They supposedly do not require any normalization. 2) They do not require any exponential computation (such as those required in sigmoid or Tanh activations).

3.4 Loss Functions

We train a network by minimizing the error the current network makes. Therefore, first, we need to define the error. The function that measures the error is called the loss function. In our experiment, we just use the regression.

Regression

For the regression tasks, we hope the predictions and targets are as close as possible. The correlation coefficient, sometimes also called the cross-correlation coefficient, is a quantity that gives the quality of a least squares fitting to the original data. It is expressed as R^2. X means the targets and the Y means the predictions. Therefore, the loss function can be any type of distance between them, like root mean squared error(RMSE), y_i means the targets and $\bar{y}_i$ means the predictions. The correlation coefficient and RMSE are given by the following equation.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \tag{2}$$

$$R^2 = \left(\frac{\mathbf{Cov}(X,Y)}{\sqrt{\mathbf{Var}(X)\mathbf{Var}(Y)}}\right)^2 \tag{3}$$

3.5 Updates

Once we have the loss function, we can update the parameters of FFNN by minimizing the error got from loss function.

Backpropagation

Backpropagation is the key algorithm that makes training deep models computationally tractable. For modern neural networks, it can make training with gradient descent as much as ten million times faster, relative to a naive implementation.

Gradient Descent

Gradient Descent The method used in conjunction with Backpropagation for finding the minimum of loss function is Gradient descent. Generates update expressions of Eq.(4). It takes steps proportional to the negative of the gradient as , because we want to minimize the loss function.

$$param = param - learningrate \times gradient \qquad (4)$$

Depending on the Size of examples in each iteration, the name will also change:

1. Stochastic Gradient Descent (SGD): one example from training set in each iteration.

2. Mini-batch gradient descent: m examples from training set in each iteration and the gradient will be averaged over m examples.

Unlike vanilla Gradient descent that runs through all samples in the training set to do a single update for a parameter in a particular iteration, SGD often converges much faster. Note that sometimes people use the term SGD even when referring to mini-batch gradient descent. The size of the minibatch is a hyperparameter but it is not very common to cross-validate it. It is usually based on memory constraints. We use powers of 2 in practice because many vectorized operation implementations work faster when their inputs are sized in powers of 2. The smaller size tends to give more generalization because it will not fit the training set too well in each iteration. Gradient Descent also has a problem. Because it is a first-order optimization algorithm that finds a local minimum of a function, it can get stuck in local minima and fail to reach the global minima as shown in Fig.2.

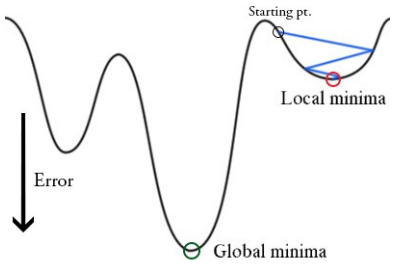
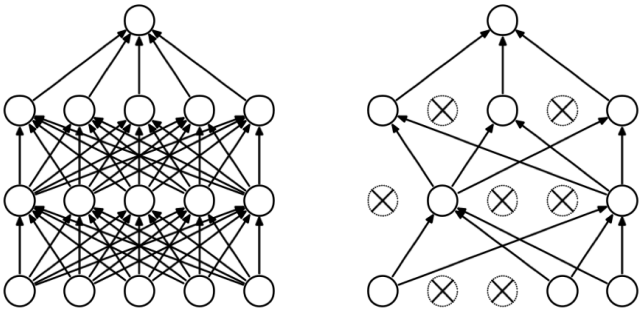


Figure 2: get stuck in local minima

The noise in the stochastic error surface is likely to bounce the network out of local minima, which is one of the reasons why SGD often converges much faster and better.

Dropout

Dropout is an extremely effective, simple regularization technique and recently introduced by Srivastava et al. In [9]. The key idea is to randomly drop units (along with their connections) from the neural network during training as indicated in Fig 3. This prevents units from coadapting too much. While training, dropout is implemented by only keeping a neuron active with some probability p (a hyperparameter), or setting it to zero otherwise.



(a) Standard Neural Net (b) After applying dropout.

Figure 3: Only apply dropout during training

4. DATASET

430 songs have been selected from Free Music Archive(FMA). Last year they filtered out the songs with low agreement to provide a cleaner development set. The extracted 45 seconds excerpts are all re-encoded to have the same sampling frequency, i.e, 44100Hz. Since at the start of the dynamic annotations the annotations were not stable, we discarded the first 15 seconds and the dynamic annotations of the last 30 seconds are provided. The 45 seconds excerpts are extracted from random(uniformly distributed) starting point a song. The dynamic (continuous) annotations were collected at a sampling rate which varied by browsers and computer capabilities. Therefore, we resampled the annotations and generated the averaged annotations with 2Hz sampling rate. To combine the annotations collected for the whole song, on nine points scale, we took the average across all annotators and rounded. The songs were annotated by crowdworkers (annotators) on Amazon Mechanical Turk. Each song was annotated once for arousal and once for valence separately. The crowdworkers were asked were asked to annotate the emotion music intends to induce and not the crowdworkers' own emotion. They had more than 1700 songs from which they selected 430 songs which had the best agreement and changes in their emotional levels. This way they provide a better set for my systems.

The database includes 430 songs. 344 songs for training and 86 songs for testing. They were labeled with arousal and valence values.The features have 260 dimensions, were extracted with openSMILE toolkit. The features include Root Mean Square energy, Zero crossings, Mel Frequency Cepstrum Coefficient, Spectral Flux, etc.

5. APPROACH AND EXPERIMENTS

Music emotion recognition system is mainly based on the study of the psychology. It can be divided into two-representation method: Categorical approach and Dimensional approach [10]. For categorical approach, SVM can be used to train a model, and then classify the categories. But there are two problems with the approach: granularity and ambiguity. Too many categories will lead to many similar categories which are nearly same from each other. However, few categories cannot lead to an effective way to distinguish different emotions. Ambiguity refers to whether the adjective emo-

tional categories used are easy or difficult to distinguish from each other. For dimensional approach, the most widely used is 2-dimensional emotion plane. The problems of granularity and ambiguity can be solved by dimensional approach. But many experiments show that the dimensions of them are not independent from each other, so this method is not perfect. In this paper, we proposed a novel deep neural network to music emotion recognition.

In Table 1, we report the performances(R^2 and RMSE) of three approaches calculated individually for each music piece. All the results are obtained by the same dataset. The analysis of the results obtained this year indicate that all our runs performed better than the algorithms of SVR and MLR. In our experiments, the results showed significant improvement over MLR and SVR. Then, we present the results of the experiments for evaluating the proposed algorithms. We use root mean square error(RMSE) and correlation coefficient(R^2) to evaluate the performance of models. The input dimension is 260 and the output dimension is 2(arousal and valence). The results are as follows:

Table 1: The performance of MLR, SVR and FFNN

Approaches	RMSE-a	RMSE-v	R^2-a	R^2-v
MLR	0.16	0.17	0.35	0.26
SVR	0.17	0.19	0.36	0.25
FFNN(hl=3)	0.25	0.26	0.68	0.66

6. CONCLUSION

In this paper, we present a novel method for emotion recognitions. The method use deep neural network to categorical approach and dimensional approach. From the numerical experiments, the method was proven to be effective for music emotion recognitions. However, there is a still considerable room to improve and show the extant distance between the human brain and the computer in our future work.

7. REFERENCES

[1] Yi-Hsuan Yang and Homer H Chen, *Music emotion recognition*, CRC Press, 2011.

[2] Moataz El Ayadi, Mohamed S Kamel, and Fakhri Karray, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern Recognition*, vol. 44, no. 3, pp. 572–587, 2011.

[3] Björn Schuller, Anton Batliner, Stefan Steidl, and Dino Seppi, "Recognising realistic emotions and affect in speech: State of the art and lessons learnt from the first challenge," *Speech Communication*, vol. 53, no. 9, pp. 1062–1087, 2011.

[4] Felix Weninger, Florian Eyben, and Björn Schuller, "On-line continuous-time music mood regression with deep recurrent neural networks," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 5412–5416.

[5] Eduardo Coutinho, George Trigeorgis, Stefanos Zafeiriou, and Björn Schuller, "Automatically estimating emotion in music with deep long-short term memory recurrent neural networks," 2015.

[6] J Stephen Downie, "The music information retrieval evaluation exchange (2005-2007): A window into music information retrieval research," *Acoustical Science and Technology*, vol. 29, no. 4, pp. 247–255, 2008.

[7] XHJS Downie, Cyril Laurier, and MBAF Ehmann, "The 2007 mirex audio mood classification task: Lessons learned," in *Proc. 9th Int. Conf. Music Inf. Retrieval*, 2008, pp. 462–467.

[8] Jürgen Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[9] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting.," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[10] Konstantin Markov, Tomoko Matsui, Francois Septier, and Gareth Peters, "Dynamic speech emotion recognition with state-space models," in *Signal Processing Conference (EUSIPCO), 2015 23rd European*. IEEE, 2015, pp. 2077–2081.

Realtime Analysis of Tweet Streams with EmoTwitter

Yuki Kobayashi Maxim Mozgovoy
The University of Aizu
Tsuruga, Ikki-machi,
Aizuwakamatsu, Fukushima, Japan
+81-242-37-2664
{m5201158, mozgovoy}@u-aizu.ac.jp

ABSTRACT

With the raise of social media, its abilities and influence on people continues to increase. It is being used for many purposes, such as communication with friends and business partners, advertisement, news delivery, information spread, and so on. To support the analysis of social media content, we developed EmoTwitter, a tool for emotion analysis of twitter messages. However, the current release of EmoTwitter supported only the analysis of a particular user's tweets, thus limiting its application area. In the present paper we outline the extension of EmoTwitter that allows realtime Twitter streams in any given geographical location.

Categories and Subject Descriptors

J.4.3 [**Computer Applications**]: Social and Behavioral Sciences – *sociology.*

General Terms

Algorithms, Design.

Keywords

Twitter, social media analysis, streaming.

1. INTRODUCTION

On the Internet, many of communication tools are used. Among them, Twitter is one of the most famous microblogs in the world, having more than 30 million active users. Twitter also became an important tool for marketing. According to research, many users follow some companies' accounts and take action that is useful for companies [1]. This trend is expected to continue in the future, so tools for twitter analysis can be of interest to sociologists and business analysts.

*ICAIT'*16, Oct. 6–8, 2016, Aizu-Wakamatsu, Japan.

EmoTwitter is a tool for emotion analysis of Twitter messages, developed at the University of Aizu [2]. It is written in C# with the support of several third-party libraries, and it supports several types of text analysis. When a user inputs a Twitter account name into a textbox, EmoTwitter displays the retrieved tweets of the given account holder. The most recent version of EmoTwitter is powered by the modern Tweetinvi library [5], and supports OAuth authentication.

However, EmoTwitter has a Twitter-imposed limit of downloading only the most recent 3200 tweets of each user, and the total hourly limit of queries. At the same time, a sociologist or a business analyst might be interested not in a particular users' messages, but in the trends in the Twittersphere in general in a particular time in a particular geographical location. The purpose of the present work is to implement such functionality.

2. EMOTWITTER IMPROVEMENTS

2.1 Streaming API and Google Maps API

Twitter provides two kinds of APIs: REST API and Streaming API. The REST API is the most basic API that allows to retrieve and analyze individual messages. The Streaming API gives the developers access to global streams of tweet data. In other words, the Streaming API is able to provide realtime twitter datastream. Figure 1 shows the comparison between REST API process and Streaming API process. Both APIs use HTTP connection, but Streaming API requires keeping a persistent HTTP connection open [3]. The Streaming API is asynchronous: the downloaded tweets are returned to the caller as they arrive, and the download process continues to work.

The Streaming API lets the user to filter the stream by geographical location and tweet language. Unfortunately, the streams available to applications only contain small extract of the real global Twitter stream. The users have to consider this limitation unavoidable.

Google Maps API for .NET is a C# interface for interacting with the backend web service of Google Maps [4]. EmoTwitter uses this API to display a Google Map object by with a web browser, search arbitrary location and get its coordinates.

2.2 Realtime Tweet Analysis

EmoTwitter is able to retrieve tweets from an arbitrary geographical area. This functionality works as follows.

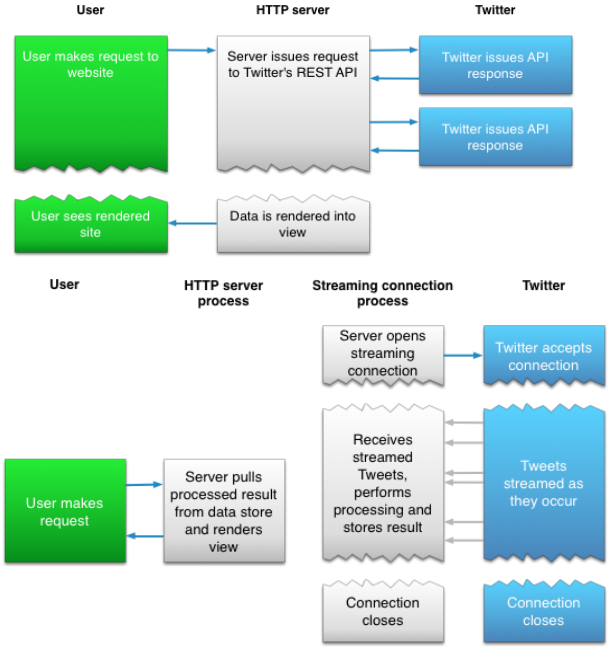


Figure 1. REST API process (upper) and Streaming API process (lower).
Source: https://dev.twitter.com/streaming/overview

First, the user inputs a location name to be analyzed. Then a Google Maps object is used to retrieve geographical coordinates to EmoTwitter. The actual search range is a square area with the corners at the following four coordinates: (latitude + 0.5 mile, longitude + 0.5 mile), (latitude + 0.5 mile, longitude – 0.5 mile), (latitude – 0.5 mile, longitude + 0.5 mile), and (latitude – 0.5 mile, longitude – 0.5 mile). Next, the user has to initiate stream analysis by clicking the "Start stream" and "Show" buttons.

EmoTwitter displays realtime tweets of the people in this location in a textbox. At the same time the most frequent words of the tweets are displayed in a word cloud (see Figure 2 and Figure 3).

3. CONCLUSION

EmoTwitter is able to retrieve realtime tweet streams of any arbitrary location in the world, providing a convenient way of accessing Twitter services for interested users. Streaming API helps to analyze ongoing trends in Twittersphere. In particular, the word cloud helps to grasp the topics under active discussion.

Our preliminary experiments show that in most cases words in a word cloud are not stable, and quickly change over time. However, sometimes it is possible to catch a somewhat longer trend where a number of keywords stay in active use for minutes and even hours. We are going to investigate this phenomenon in further experiments.

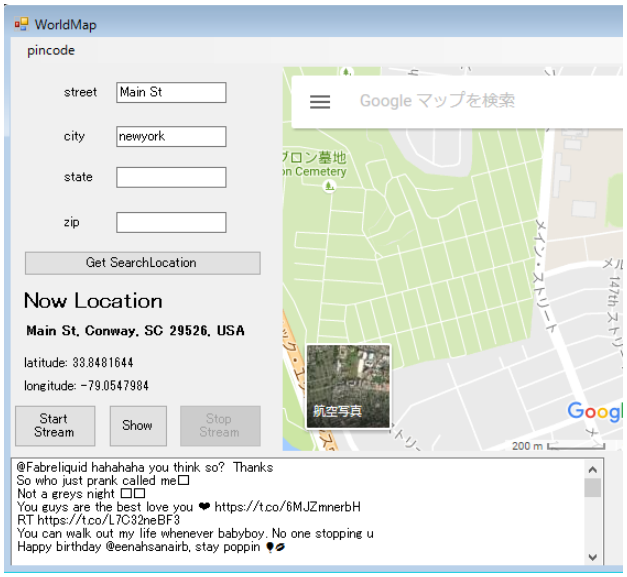


Figure 2. Choosing a location around New York Main St.

Figure 3. Example word cloud of a realtime stream.

4. REFERENCES

[1] Twitter ユーザー調査 2016: https://cdn.cms-twdigitalassets.com/content/dam/business-twitter/resources/customerinsightsloc/Customer_insights_2016_EN_06.28.2016_JA.pdf.

[2] M. Munezero, C. Montero, M. Mozgovoy, E. Sutinen. EmoTwitter — A Fine-Grained Visualization System for Identifying Enduring Sentiments in Tweets. *Lecture Notes in Computer Science*, 2015, vol. 9042, pp. 78-91.

[3] The Streaming APIs Overview. https://dev.twitter.com/streaming/overview.

[4] Google Maps API for .NET. https://github.com/ericnewton76/gmaps-api-net.

[5] Tweetinvi homepage: https://github.com/linvi/tweetinvi.

Analysis of Emoticons in a Japanese Twitter Corpus

Hiroshi Yamaguchi
The University of Aizu
Tsuruga, Ikki-machi,
Aizuwakamatsu, Fukushima, Japan
+81-242-37-2664
m5201115@u-aizu.ac.jp

Maxim Mozgovoy
The University of Aizu
Tsuruga, Ikki-machi,
Aizuwakamatsu, Fukushima, Japan
+81-242-37-2664
mozgovoy@u-aizu.ac.jp

ABSTRACT

This paper describes relation of emoticon and emotion in the text. We used a Japanese Twitter corpus that was annotated with eight emotions based on Ekman and Plutchik emotion categories. We analyze the relation between emoticons and emotions, and briefly discuss obtained results. We are convinced that this work will be useful for researcher interested in emotion analysis of the texts.

Categories and Subject Descriptors

I.2.7 [**Natural Language Processing**]: Text analysis.

General Terms

Algorithms.

Keywords

Twitter, emotions.

1. INTRODUCTION

Emoticons express emotion of a writer texts. We use emoticons in various online contexts, such as e-mailing, chats, and social networking services. An emoticon consists of a string of characters and punctuation symbols such as "(´ ; ω; `)". There is a large number of emoticons used in the Internet for communication, and some emoticons may correspond to several emotions.

In this research we use a Japanese Twitter corpus annotated with six emotions, suggested by Ekman [2] and Plutchik [3]: "anger", "disgust", "sadness", "surprise", "fear" and "happiness", and two two extra emotions, "pride" and "embarrassment" [1]. We manually created the list of all emoticons found in this corpus to establish the link between emotions and emoticons.

Our corpus consists of 4338 tweets. 2215 of them (51.0%) are marked as having emotional value, and 791 of them (18.2%) are marked as containing emoticons. Emotions in the corpus are distributed very unevenly. For example, the most frequent emotion is "happiness" comprising nearly 40% in this corpus, whereas the least frequent emotion is "fear" comprising nearly 2%.

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

2. EMOTICON ANALYSIS

To establish the link between emotions and emoticons, we manually created the list of all emoticons found in our corpus. We counted how many times each emotion is marked with a particular emotion. For example if a certain emoticon is marked with the emotion of happiness and pride, it gets one "happiness score" and "pride score". When the same emoticon appears many times, its score is increased. If the same emoticon appears having another emotion, it will also receive the score of that emotion category.

Next, we calculated emotion percentage, showing the range of emotions associated with a particular emoticon, as follows:

$$Emotion\ percentage\ = \\ Single\ emotion\ score\ /\ \ Total\ emotion\ score$$

A score of a single emotion is one of the eight emotion scores such as "happiness score" or "pride score". Total emotion score is the total score of each emoticon. In other words, the sum of "happiness score", "pride score" and so on. The emotion percentage describes the weighted range of emotions that each emoticon may have. It is useful when we discuss the links between individual emotions and emoticons. To understand the contexts of emoticon use in texts, we performed the following process. First, we calculated normalized emotion percentages:

$$Normalized\ percentage\ of\ emotion\ = \\ Largest\ percentage\ among\ emotions\ / \\ Percentage\ of\ emotion$$

For each emoticon we then calculate a ratio between the highest and the second highest normalized percentage of emotion. The resulting value represents emotional confidence showing the degree of ambiguity of each emoticon. If the ratio is high, it means the emoticon is used almost always in the same emotional context. If the ratio is low, it means that the emoticon is widely used in different contexts, so the emoticon tends to be ambiguous.

The Table 1 shows emotional confidence values of the most frequent emoticons in the corpus. The most frequent emotion is almost always "happiness", but the second frequent

emotion change greatly across emoticons.

The Table 2 describes ten emoticons with the lowest emotional confidence values. Again, the most probable emotion is "happiness".

Table 1: Emotional confidence values for most frequent emoticons in the corpus

Emoticon	Fre	Largest	Second	Ratio
\(^o^)/	40	happiness	surprise	1.9
^^	39	happines	pride	2.9
(^o^)	30	happiness	pride	6.0
(´ ▽ `) ノ	28	happiness	pride	6.2
(^^)	23	happiness	pride	5.8
(^O^)/	21	happiness	sadness, pride	8.3
(´ ・ω・ `)	20	sadness	surprise	4.4
(^-^)	17	happiness	surprise	7.7
(´ Д `)	16	happiness	pride	6.0
(^ω^)	15	happiness	surprise	2.1

Table 2: Emoticons with the lowest emotional confidence

Emoticon	Largest	Second	Ratio
(` ・ω・ ´)	happiness	pride	1.2
><	sadness	happiness	1.2
(-_-)	sadness	happiness	1.3
ヽ (´ ▽ `)/	happiness	sadness	1.3
f^_^;)	happiness	surprise	1.3
(´ Д ` ;)	surprise	sadness	1.5
(^o^ ゞ	happiness	pride	1.5
(((o(*゜▽゜*)o)))	pride	happiness	1.5
^^;	sadness happiness pride embarrassment	disgust	1.5
((((;゜Д ゜))))))))	sadness	surprise	1.5

3. DISCUSSION

The results of corpus analysis are summarized in the Table 1 and Table 2. If a certain emoticon has a low emotional confidence value, it does not correspond to a specific emotion. Table 2 provides examples of such emoticons that can be used in a variety of emotional contexts.

"Happiness" is the most used emotion in the corpus. "Happiness" is a broad emotional category that is related to "joy", "satisfaction", "love" and other positive emotions, which can explain high frequency of "happiness".

In the Table 2, the most ambiguous emoticon "(` ・ω・ ´) " represents both "happiness" and "pride". Both emotions are positive, so this result does not necessarily mean ambiguity.

Table 3: Emoticons with the highest emotional confidence

Emoticon	Largest	Second	Ratio
(´ ; ω ; `)	sadness	happiness	12.5
(^^)/	happines	pride	11.0
(^▽^)	happiness	anger	9.0
(^O^)/	happiness	sadness, pride	8.3
(^-^)	happiness	surprise	7.6
(´ ▽ `)	happiness	anger, pride	7.5
(* ´ ∀ ` *)	happiness	surprise	7.5
(#^-^#)	happiness	pride	7.0
(*^^*)	happiness	surprise	6.5
(´ ▽ `) ノ	happiness	pride	6.2

However, the second ambiguous emoticon "><" represents both "sadness" "happiness". These are completely opposite emotions, so the given emoticon can be used in different contexts.

Unfortunately, most of ambiguous emoticons occur rarely in the corpus, so it is difficult to judge whether they are truly ambiguous or their emotional scores are unreliable. One interesting result is "^^;". This emoticon has high emotional confidence values of "sadness", "happiness", "pride" and "embarrassment". This emoticon can be widely used in both positive and negative contexts.

4. CONCLUSIONS

In this research, we discussed the relation between emoticons and emotions in the text. Some emoticons are used for different emotional meanings in texts. Sometimes an emoticon may represent opposite emotions, depending on the context. Emoticons do not necessarily represent emotions originally associated with them. Furthermore, emoticons complement the emotions not reflected in the words.

The results of this research are quite unbalanced due to biased emotional categorization. Especially, there are nearly 40% of "happiness"-marked emoticons in our corpus. In the future, we will need re-mark the corpus to get a more balanced emotional classification and thus obtain more reliable results.

5. REFERENCES

[1] A. Danielewicz-Betz, H. Kaneda, M. Mozgovoy, and M. Purgina, " Creating English and Japanese Twitter corpora for emotion analysis " , International Journal of Knowledge Engineering, 2015, vol. 1(2), pp. 120-124.

[2] P. Ekman, " Basic Emotions, " in Handbook of Cognition and Emotion, T. Dalgleish and M. Power, Eds., John Wiley & Sons, 1999, pp. 45-60.

[3] R. Plutchik, " The nature of emotions, " American Scientist, vol. 89, no. 4, 2001, pp. 344-350.

Multiprogrammed Control Problem of One Supply Chain

Inna Trofimova
St. Petersburg State University,
7/9, Universitetskaya Embankment,
St.Petersburg, Russian Federation, 199034 and
St.Petersburg institute for informatics and
automation of the Russian academy of sciences
39, 14 Line, St.Petersburg, 199178, Russia
isolovyeva@mail.ru

Larisa Mynzu
St. Petersburg State University,
7/9, Universitetskaya Embankment,
St.Petersburg, Russian Federation, 199034
larisabarisa@gmail.com

ABSTRACT

In this paper the problem of construction of the multiprogrammed control for desired (opportune) programmed regimes is considered. To demonstrate proposed methods, we applied them to the model of one supply chain. Results are verified with the help of this procedure realization on numerical examples.

Categories and Subject Descriptors

PBUH [**Optimization**]: Linear Programming; PBWH [**Mathematics**]: Mathematical Modelling

General Terms

Mathematical model of supply chain, control problem

Keywords

Multiprogrammed control, positional optimization method, supply chain

1. INTRODUCTION

Nowadays growth of global market causes progressive improvement of operating in modern supply chains (SC). Accomplish this SC strive to work with a large variety of products, to achieve a high quality supply and a reliability and environmental standards, to take into account fast appearance of new products and increasing competitiveness of the companies, to apply information technologies. In other words today's market is like a highly dynamic environment and accordingly this supply chains should correspond highly dynamic systems. Different mathematical models for describing SC dynamic behavior were introduced in recent times. Part of them is used for optimization of operating SC and application of control theory methods for solving mentioned problems, for example [9],[6],[8].

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

Many companies primarily take into account cost saving and compliance with deadlines when unforeseen events happen. For example these events could occur due to transfer and storage of products. Accordingly this they attempt to analyze ability of their system to react on time and to manage SC in this cases.

Modern technical capacity of receiving information about SC current condition enable to improve SC operating. Thus one of the main challenges in the SC management is SC execution problem in the case of external disturbances [3], [2].

In this paper the problem of construction of the multiprogrammed control for desired (conducive) programmed regimes is considered. We suggest to use positional optimization method and linear programming methods for solving this problem. To demonstrate proposed methods, we applied them to the model of one supply chain.

2. MODEL DESCRIPTION

Let us consider the multiprogrammed control problem of one SC. Following [1], we consider a mathematical model of generalized supply chain functioning.

It is assumed that supply chain consist of three elements: manufacturer (plant), a warehouse and the customer (Figure 1). Under the plant we mean one or more plants of the same manufacturer, which produced only one type of products manufactured with the necessary power. This product comes from the factory to the warehouse, and from there It delivered to the customer with the corresponding demand level. Consumer here can be a wholesale manufacturer, dealer companies and end consumers. It is assumed that the work of the supply chain changes dynamically.

In this paper, using this model [1], we'll find the optimal production rate that provides a predetermined level of demand, taking into account incoming information about inventory level in the warehouse. $I(t)-$the inventory level at

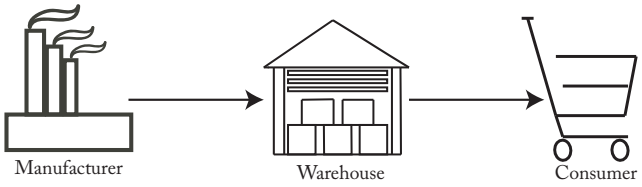


Figure 1: Operational scheme of considered supply chain

time t, $P_a(t)-$the actual production rate at time t, $D(t)-$the demand rate at time t, $u(t)-$the desired production rate at time t, $\theta(t)-$the deterioration rate, $\alpha-$the inverse of exponential delay time. All functions are assumed to be nonnegative, continuous and differentiable. Therefore first order differential equations of considered model are following:

$$\begin{cases} \dot{I}(t) = P_a(t) - \theta(t)I(t) - D(t) \\ \dot{P}_a(t) = \alpha(t)(u(t) - P_a(t)) \end{cases} \quad (1)$$

Let us introduce the following notation:

$$\frac{d}{dt}\begin{pmatrix} I \\ P_a \end{pmatrix} = \begin{pmatrix} -\theta & 1 \\ 0 & -\alpha \end{pmatrix}\begin{pmatrix} I \\ P_a \end{pmatrix} + \begin{pmatrix} 0 \\ \alpha \end{pmatrix}u(t) + \begin{pmatrix} -D(t) \\ 0 \end{pmatrix},$$

$$\begin{pmatrix} I \\ P_a \end{pmatrix} = x(t), \begin{pmatrix} -\theta & 1 \\ 0 & -\alpha \end{pmatrix} = A(t), \begin{pmatrix} 0 \\ \alpha \end{pmatrix} = b(t), \begin{pmatrix} -D \\ 0 \end{pmatrix} = f(t).$$

For brevity sake (1) could be rewrite in the form:

$$\dot{x}(t) = A(t)x(t) + b(t)u(t) + f(t), \quad (2)$$

where $A(t) - n \times n$ matrix; $b(t) - n$-dimensional vector function; $u(t)-$the value of control action; $f(t)-$piecewise continuous bounded disturbance.

The model parameters could be defined by identifying the model using the statistical information about demand, output volumes and the inventory level on the previous periods of time of SC work. For example, the products arrival rate from the warehouse to the consumer could be received from the statistical information about turnover of a warehouse and the estimates of demand for products for the previous period. The input parameters of the model are defined with a certain accuracy, so the results of numerical experiments for different values of the parameters could help the manufacturer to decide the value of the release.

3. PROBLEM STATEMENT

Let us consider the supply chain (Fig. 1), consisted of the manufacturer, consumer and warehouse. Assume that the desired programmed regimes (schedule of SC operation) $x_i(t)$, $i = \overline{1,n}$, for system (2) are provided with programmed controls $u_{pi}(t)$, $i = \overline{1,n}$. To realize n different desired programmed regimes x_i we can construct a multiprogrammed control [**?**]

$$u_{mp}(x(t),t) = \sum_{i=1}^{n} u_{p_i}(t) \prod_{j \neq i} \frac{(x(t) - x_j(t))^2}{(x_i(t) - x_j(t))^2}, \quad (3)$$

where scalar product is denoted as $(x_i(t) - x_j(t))^2$. The main property of this multiprogrammed control (3) is $u_{mp}(x_i(t),t) = u_{p_i}(t)$, $i = \overline{1,n}$.

Also suppose that the manufacturer has the opportunity to obtain information about the level of inventory in stock and compare it with the actual rate of production at fixed moments of time. At these moments, the manufacturer can change the production strategy (control the value of release) to provide the expected level of demand and maximize profits at the end of the planning period of time. Since the planned volume of supply may differ from current, the manufacturer has to regularly adjust the work of the supply chain. We assume that the control u belongs to the class of piecewise constant functions, that is, the manufacturer can change the volume of release (control u) at certain times. Thus, the aim is to find an optimal control u of the SC described by the system (2), providing the expected level of demand at the moment of time t^*, given by constraints: $Hx(t^*) = g, g \in R^m, rankH = m < n$ and maximizing profits, described by $c'x(t) \to max$. Furthermore, another our goal is to conduct numerical experiments for different values of the parameters.

4. METHODS

Let us design optimal control: $u(x(t),t) = u_{mp}(x(t),t) + v(t)$. At first for each programmed regime $x_i(t), i = \overline{1,n}$ vectors $y_i(t) = x(t) - x_i(t)$, $i = \overline{1,n}$ could be considered and a deviation system for $i = \overline{1,n}$ could be obtained. In view of multiprogrammed control (3) each of them is nonlinear system.

In this study we propose to use Positional optimization method [7] for solving the SC multiprogrammed control problem. This method is based on widely known mathematical methods of optimal control problems and linear programming (adaptive method) [5]. It enables to solve control problems of both linear and nonlinear systems and it was applied in a number of cases, for example [7], [10], [4]. Adaptive method algorithms offers variety of advantages, because it is finiteness, exact and relaxation, algorithm. Moreover one important fact that it focuses on the linear programming problems with banded structure of matrix, that are appear when there are a large number of segments N of considered time interval $[t_*, t^*]$. Here it permits us to get a current value of positional solution and calculate $v(t)$ in control process.

The main idea of this method resides in sequential designing of optimal programmed control (OPC) on the time interval T. In this process current entry information about system state is taken into consideration at each period of time. Let $T = [t_*, t^*]$, $h = (t_* - t^*)/N$, $t_* < t^* < +\infty$ $T_u = \{t_*, t_* + h, \ldots, t^* - h, t^*\}$, function $v(t)$, $t \in T$ define as a discrete control, if $v(t) = v(t_* + kh)$, $t \in [t_* + kh, t_* + (k+1)h]$, $k = \overline{1, N-1}$. In this class of discrete controls we consider the auxiliary OPC problem with one of achieved deviation systems at the moment of time $\tau = t_* + kh$ on the time interval $T_\tau = \{\tau, \tau + h, \ldots, t^* - h, t^*\} \in T_u$.

The solution of the auxiliary OPC problem can be divided into two steps: the reduction of an optimal control problem for interval linear programming problem and solving interval linear programming problem by adaptive method.

5. NUMERICAL EXAMPLES

Above described procedure was realized in MATLAB. It was verified on the systems with not large dimensions. For the SC with input parameters from Table 1 we calculate optimal positional control $v(t)$. It is demonstrated on the Figure 2.

Table 1: Input Parameters

Input Parameters						
A		b	f	c	t_*	t^*
-0.05	1	0	$-2\cos t_i$	-2	0	π
0	-5	5	0	2		

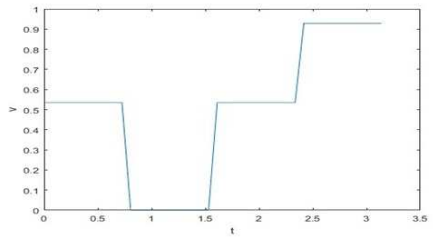


Figure 2: Values of optimal positional control $v(t)$ in numerical example

6. CONCLUSION

In this paper the problem of designing multiprogrammed control of one SC is considered. Here we propose the approach to design optimal control of one SC with the availability several early picked desired regimes and in case it is subject to external actions. Results are verified with the help of this procedure realization on numerical examples. In prospect we are going to apply this solution strategy to the model of real logistics or manufacturer company and to test it on real data. Additionally, we are eager to compare results of our modeling with the results obtained using other methods of SC control on real data.

7. ACKNOWLEDGMENTS

The research described in this paper is partially supported by the Russian Foundation for Basic Research (grants 15-07-08391, 15-08-08459, 16-07-00779, 16-08-00510, 16-08-01277, 16-29-09482), grant 074-U01 (ITMO University), project-6.1.1 (Peter the Great St.Petersburg Politechnic University) supported by Government of Russian Federation, Program STC of Union State "Monitoring-SG" (project 1.4.1-1), State researches 0073-2014-0009, 0073-2015-0007.

8. REFERENCES

[1] F. Bukhari. Adaptive control of a production-inventory model with uncertain deterioration rate. *Applied Mathematics*, 2:1170–1174, 2011.

[2] CA. Garcia, A. Ibeas. Inventory control for the supplychain: An adaptive control approach based on the identification of the lead-time. *Omega*, 40:314–327, 2012.

[3] D. Schwartz, W. Wang. Simulation-based optimization of process control policies for inventory management in supply chains. *Automatica*, 125(2):1311–1320, 2006.

[4] D.A. Ivanov, B.V. Sokolov. Application of control theoretic tools to supply chain disruptions management. In *Proceedings of 7th IFAC Conference on Manufacturing Modelling, Management, and Control*, pages 1926–1931. IFAC Publisher, 2013.

[5] R. Gabasov and F. Kirillova. *Linear programming methods. Part 3. Special problems.* Librocom, Russia, Moscow, 2010.

[6] K. Subramanian, J. B. Rawlings. Integration of control theory and scheduling methods for supply chain management. *Computer and Chemical Engineering*, 51:4–20, April 2013.

[7] N.V. Balashevich, R. Gabasov and F. Kirillova. Optimal control of nonlinear systems. *Computational Mathematics and Mathematical Physics*, 42(7):931–956, 2002.

[8] M. Ortega. Control theory applications to the production-inventory problem: a review. *International Journal of production Research*, 8(2):74–80, 2000.

[9] E. Perea, I. Grossman, E. Ydstie, and T. Tahmassebi. Dynamic modeling and classical control theary for supply chain management. *Computer and Chemical Engineering*, 24:1143–1149, July 2000.

[10] N. V. Smirnov. Multiprogram control for dynamic systems: A point of view. In *ACM International Conference Proceeding Series, Joint International Conference on Human-Centered Computer Environments, HCCE 2012*, pages 106–113, 2012.

Implementation and Testing of the Fast Numerical Algorithm for Simulation of 3D Gravity Creeping Flow of Incompressible Newtonian Fluid

Timofey V. Abramov
Trofimuk Institute of Petroleum Geology and Geophysics of Siberian Branch of Russian Academy of Sciences
Koptug ave. 3, 630090
Novosibirsk, Russia
AbramovTV@ipgg.sbras.ru

Mikhail M. Lavrentiev
Novosibirsk State University
Pirogova Str. 2, 630090
Novosibirsk, Russia
mmlavr@nsu.ru

Boris V. Lunev
Trofimuk Institute of Petroleum Geology and Geophysics of Siberian Branch of Russian Academy of Sciences
Koptug ave. 3, 630090
Novosibirsk, Russia
LunevBV@ipgg.sbras.ru

ABSTRACT

A method for fast numerical simulation of 3D gravitational creeping flow of uniform viscous incompressible Newtonian fluid with piecewise constant density in a half-space bounded above by a free surface is presented. The method is based on the known explicit form of Green's function of corresponding boundary value problem.

As it is known, the solution to this problem may be found just as a convolution of two known functions — Green's one and right hand side of the equation, instead of solving a difference equation. Direct algorithm requires $O(N^2)$ operations to calculate velocity filed in the entire half-space, where N is the number of nodes of the computational grid. Obviously, this procedure is extremely inefficient when N is large enough (about 10^6 and more), and parallel computing does not resolve the problem.

In this paper the modification of the calculation algorithm is proposed, implemented and compared with the direct one. The modification consists in the conversion of the desired convolution-like sum (solution to the problem) to the form of cyclic convolution, which can be calculated in $O(N \log N)$ operations by a fast algorithm.

Categories and Subject Descriptors

F.2.1 [**Analysis of algorithms and problem complexity**]: Numerical Algorithms and Problems—*Computation of transforms*; G.1.0 [**Mathematics of Computing**]: Numerical analysis—*Numerical algorithms, Parallel algorithms*; G.1.8 [**Mathematics of Computing**]: Partial Differential Equations—*Elliptic equations*

Keywords

Rayleigh-Taylor instability, Newtonian fluid, boundary value problem, Green's function, cyclic convolution, fast Fourier transform

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

1. INTRODUCTION

Gravitational creeping flow of inhomogeneous (primarily, with different densities) incompressible Newtonian fluid is widely used as a mathematical model for many geological processes in the Earth. In such way evolution of sedimentary basins, thermal convection in the mantle, continents' influence on the heat flows in the mantle and many other processes are simulated. Perhaps, the most practical importance has salt tectogenesis (salt diapirism) — solid rocks upwelling through more dense sediments because of low density of salt, which creates high-amplitude mushroom-shaped structures (salt diapirs or domes, see Fig. 1).

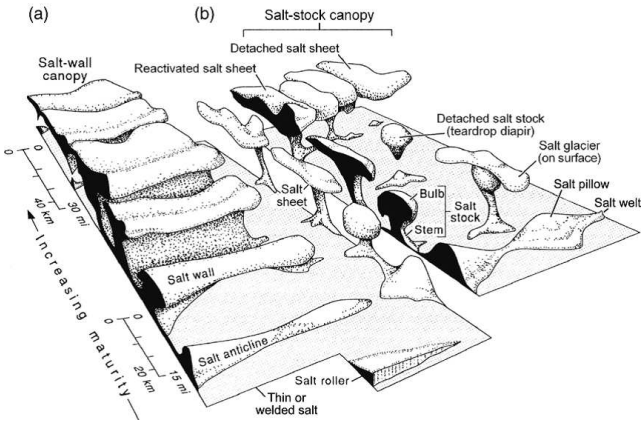


Figure 1: Diagram observed various salt structures. Structural maturity and size increase toward the background. (a) Structures rising from line sources. (b) Structures rising from point sources [3].

The practical interest of geologists and geophysicists is in the fact that salt tectogenesis appears in almost all major oil and gas provinces, it complicates the structure of the sedimentary basin, so it determines the distribution of hydrocarbon deposits [9].

According to modern understanding, physically it is a spe-

cial case of Rayleigh-Taylor instability and it is adequately described by gravitational creeping flow of incompressible Newtonian fluid[1].

To study a physical process via numerical simulation it is necessary to make a great number of computational experiments, varying the model parameters. Therefore, fast simulation tools are on demand. Fortunately, there is an analytical solution of the corresponding boundary value problem for gravitational creeping flow of Newtonian fluid with inhomogeneous density and continuous viscosity, which is bounded by a free surface [5, 6]. The corresponding formulae make it possible to create several efficient algorithms for calculation of such flows.

In this paper we describe 3D fast numerical algorithm for solving the considered problem. Precise mathematical statement of the problem, as well as the corresponding discrete version, are given in Section 2. Section 3 describes the proposed fast numerical algorithm, while the results of numerical tests are presented in Section 4.

2. MATHEMATICAL STATEMENT

Consider a half-space bounded above by a free surface, occupied by several incompressible and immiscible Newtonian fluids with very high uniform viscosity ($\sim 10^{20}$ Pa $\cdot$ s) and different densities, as shown in Fig. 2. The problem is to determine the gravity creeping flow of these fluids with $\mathbf{g} = (-g, 0, 0)$.

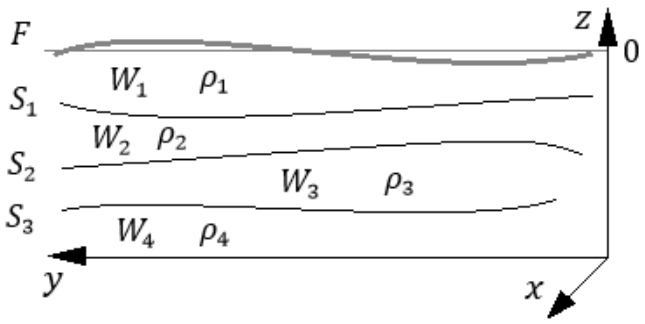


Figure 2: A half-space in Cartesian coordinate system $\mathbf{x} = (x, y, z)$ occupied by liquids $W_1, \ldots, W_4$ with densities $\rho_1, \ldots, \rho_4$, respectively, and uniform viscosity μ. They are separated by the surfaces $S_1, \ldots, S_3$, and bounded above by the free surface F.

Each time moment t the density distribution in the half-space is uniquely determined by the actual configuration of surfaces S_i: $\rho(\mathbf{x}, t) = \rho_i$ for $\mathbf{x} \in W_i$. Density ρ, stress tensor $\mathbf{T}$ and pressure P are represented as:

$$\begin{aligned} \rho(\mathbf{x}, t) &= \rho^0(z, t) + \sigma(\mathbf{x}, t) \\ \mathbf{T}(\mathbf{x}, t) &= \mathbf{T}^0(z, t) + \boldsymbol{\tau}(\mathbf{x}, t) \\ P(\mathbf{x}, t) &= P^0(z, t) + p(\mathbf{x}, t), \end{aligned} \tag{1}$$

where $\rho^0(z, t)$, $\mathbf{T}^0(z, t)$, $P^0(z, t)$ characterize hydrostatic condition ($\mathbf{T}^0(z, t) = -\delta_{ij} P^0(z, t) = -\delta_{ij} g \int_0^z \rho^0(z, t) dz$, where δ_{ij} — Kronecker symbols), and $\sigma(\mathbf{x}, t)$, $\boldsymbol{\tau}(\mathbf{x}, t)$ $p(\mathbf{x}, t)$ — small deviations.

As it is shown in [5, 6], the problem of determining a creeping flow may be formulated with respect to the deviations and splitting into quasi-stationary part and evolution part. The quasi-stationary part with linearized boundary conditions has the form:

$$\begin{aligned} &\mu \boldsymbol{\nabla}^2 \mathbf{v} - \boldsymbol{\nabla} p = -\sigma \mathbf{g}, \\ &\boldsymbol{\nabla} \cdot \mathbf{v} = 0, \\ &(v_z = \tau_{zx} = \tau_{zy} = 0) \mid_{z=0}, \end{aligned} \tag{2}$$

with the additional condition for determining $\zeta(x, y)$ — perturbations of free surface $F = z - \zeta(x, y) = 0$:

$$(\tau_{zz}) \mid_{z=0} = -\rho^0 |\mathbf{g}| \zeta. \tag{3}$$

The evolution part is as follows

$$\frac{\partial S_i}{\partial t} + \mathbf{v} \cdot \boldsymbol{\nabla} S_i = 0. \tag{4}$$

Creeping flow at every time moment $t^{(k)}$ is evaluated in two stages. Firstly, we obtain velocity field $\mathbf{v}^{(k)}$, which is determined by $\rho(\mathbf{x}, t_k)$, i. e. by the configuration of surfaces $S_i^{(k)}$. Then the evolution equation (4) is solved, i. e. surfaces $S_i^{(k)}$ are moved by $\mathbf{v}^{(k)}$ to the next state $S_i{}^{(k+1)}$. New configuration of surfaces $S_i^{(k+1)}$ determines new density distribution $\rho(\mathbf{x}, t_{k+1})$, which provides new velocity field $\mathbf{v}^{(k+1)}$. Thus, the evolution of all layers are described by a sequence of steady states, each subsequent element of which is connected with the previous one through the evolution of surfaces S_i.

For equation (2) the analytical form of Green's function was found in [5], that allows to find $\mathbf{v}$ just as a convolution of this function with density distribution, without using any finite difference methods [1, 6]:

$$\mathbf{v}(\mathbf{x}) = \iiint \mathbf{V}(\mathbf{x}, \boldsymbol{\xi}) \sigma(\boldsymbol{\xi}) d\xi_x d\xi_y d\xi_z, \tag{5}$$

where $\mathbf{V} = (V_x, V_y, V_z)$ — Green's function, $\mathbf{x}$, $\boldsymbol{\xi}$ — Cartesian coordinates. Discrete version of (5) is nothing but:

$$\mathbf{v}(\mathbf{n}) = h_x h_y h_z \sum_{m_x=0}^{N_x-1} \sum_{m_y=0}^{N_y-1} \sum_{m_z=0}^{N_z-1} \mathbf{V}(\mathbf{n}, \mathbf{m}) \sigma(\mathbf{m}) \tag{6}$$

where $\mathbf{n} = (n_x, n_y, n_z)$ and $\mathbf{m} = (m_x, m_y, m_z)$ are discrete coordinates; $n_x = 0, \ldots, N_x - 1$, $n_y = 0, \ldots, N_y - 1$, $n_z = 0, \ldots, N_z - 1$; N_x, N_y, N_z — the size of uniform grid of density distribution; and h_x, h_y, h_z are the grid steps in different directions.

Evaluation of the velocity field $\mathbf{v}$ by (6) has very high computational cost. Direct algorithm requires $O(NM)$ operations, where N — the number of nodes of the density grid and M — the number of points, in which velocity vector is evaluated. To solve equation (4) we need to determine $\mathbf{v}$ field only at surfaces S_i, so, $M \ll N$. However, sometimes it is necessary to get $\mathbf{v}$ on the entire regular grid and $M = O(N)$. In both cases, using the analytical solution of this problem far more preferable than using finite difference methods, which demand $O(N^3)$ operations [4].

However, $O(N^2)$ and even $O(NM)$ is quite high complexity, that makes it impossible to increase significantly the size of the problem, which is important for applications. So, the fast algorithm has been used for evaluation boundary value

[1]This work is focused on salt diapirism and it is a continuation of a series of earlier authors' studies on the subject [1, 6, 7, 2]. However, the presented results can be generalized to a wider class of problems

problems with known explicit form of Green's function (i. e. evaluation expressions like (6)), proposed in [2].

3. FAST NUMERICAL ALGORITHM

One of possible optimization methods consists in using fast convolution algorithms [8], which allow to calculate *discrete cyclic convolution* by $O(N \log N)$ operations instead of $O(N^2)$ operations for direct algorithm. In 3D space it is defined as follows:

$$y(\mathbf{n}) = f(\mathbf{n}) * g(\mathbf{n}) = \\ = \frac{1}{N_x}\frac{1}{N_y}\frac{1}{N_z}\sum_{m_x=0}^{N_x-1}\sum_{m_y=0}^{N_y-1}\sum_{m_z=0}^{N_z-1} f(\mathbf{n}-\mathbf{m})g(\mathbf{m}) \quad (7)$$

where $f(\mathbf{n})$ and $g(\mathbf{n})$ — convolved functions of discrete variables; $n_x = 0, \ldots, N_x - 1$, $n_y = 0, \ldots, N_y - 1$, $n_z = 0, \ldots, N_z - 1$. Functions $f(\mathbf{n})$ and $g(\mathbf{n})$ are *assumed to be periodic* with period $\mathbf{N}$, i. e. $f(\mathbf{n}) = f(k\mathbf{N} + \mathbf{n})$ and $g(\mathbf{n}) = g(k\mathbf{N} + \mathbf{n})$.

Such function can be evaluated fast by well-known algorithms. However, expression (6) is not equal to (7), so we are not able to use these algorithms directly.

The way to overstep this problem was proposed in [2]. First, in our case Green's function of six variables $\mathbf{V}(\mathbf{x}, \boldsymbol{\xi})$ is regarded as the function of three ones: $\mathbf{V}(\mathbf{x} - \boldsymbol{\xi})$. Assume that it is the function of discrete variables: $\mathbf{V}(\mathbf{n} - \mathbf{m})$, where $n_x, m_x = 0, \ldots, N_x - 1$, $n_y, m_y = 0, \ldots, N_y - 1$, $n_z, m_z = 0, \ldots, N_z - 1$. Let us consider the new function $\hat{\mathbf{V}}(n_x, n_y, n_z)$ according to:

$$\hat{\mathbf{V}} = \begin{cases} \mathbf{V}(n_x, n_y, n_z), \\ \qquad n_x < N_x, n_y < N_y, n_z < N_z; \\ \mathbf{V}(n_x - 2N_x + 1, n_y, n_z), \\ \qquad n_x \geq N_x, n_y < N_y, n_z < N_z; \\ \mathbf{V}(n_x, n_y - 2N_y + 1, n_z), \\ \qquad n_x < N_x, n_y \geq N_y, n_z < N_z; \\ \mathbf{V}(n_x, n_y, n_z - 2N_z + 1), \\ \qquad n_x < N_x, n_y < N_y, n_z \geq N_z; \\ \mathbf{V}(n_x - 2N_x + 1, n_y - 2N_y + 1, n_z), \\ \qquad n_x \geq N_x, n_y \geq N_y, n_z < N_z; \\ \mathbf{V}(n_x - 2N_x + 1, n_y, n_z - 2N_z + 1), \\ \qquad n_x \geq N_x, n_y < N_y, n_z \geq N_z; \\ \mathbf{V}(n_x, n_y - 2N_y + 1, n_z - 2N_z + 1), \\ \qquad n_x < N_x, n_y \geq N_y, n_z \geq N_z; \\ \mathbf{V}(n_x - 2N_x + 1, n_y - 2N_y + 1, n_z - 2N_z + 1), \\ \qquad n_x \geq N_x, n_y \geq N_y, n_z \geq N_z; \end{cases} \quad (8)$$

where $\hat{\mathbf{V}}(n_x, n_y, n_z)$ is defined on 3D uniform grid of $(2N_x - 1) \times (2N_y - 1) \times (2N_z - 1)$ nodes. Let us also define a new function $\hat{\sigma}(n_x, n_y, n_z)$ on a grid of the same size:

$$\hat{\sigma} = \begin{cases} \sigma(n_x, n_y, n_z) & , n_x < N_x, n_y < N_y, n_z < N_z; \\ 0 & , otherwise. \end{cases} \quad (9)$$

As it was shown in [2], cyclic convolution of $\hat{\mathbf{V}}(\mathbf{n})$ ($\hat{V}_x, \hat{V}_y, \hat{V}_z$ are treated separately) and $\hat{\sigma}(\mathbf{n})$ provides exact (up to a constant factor) solution to equation (6):

$$\begin{aligned} &\hat{V}_x(\mathbf{n}) * \hat{\sigma}(\mathbf{n}) = v_x(\mathbf{n}), \\ &\hat{V}_y(\mathbf{n}) * \hat{\sigma}(\mathbf{n}) = v_y(\mathbf{n}), \\ &\hat{V}_z(\mathbf{n}) * \hat{\sigma}(\mathbf{n}) = v_z(\mathbf{n}), \quad \text{where :} \\ &n_x = 0, \ldots, N_x - 1; \\ &n_y = 0, \ldots, N_y - 1; \\ &n_z = 0, \ldots, N_z - 1. \end{aligned} \quad (10)$$

These define the fast $O(N \log N)$ algorithm for calculating stationary velocity field, so the overall computational complexity of creeping flow simulation is $O(N \log N)$, too.

4. RESULTS

The fast algorithm, described in this paper, was implemented and tested. For fast calculation of discrete cyclic convolution the classical algorithm, based on the convolution theorem [8], was used. Fast Fourier transforms were calculated via cuFFT program library on GPU and via multithreaded FFTW library on CPU.

The direct algorithm for (6) using GPU computing via NVIDIA® CUDA® technology has been also implemented by the author [1]. The efficiency of this implementation has been verified by the NVIDIA's profiler `nvprof`, and it has shown compute utilization about 80–95 %[2], i. e. the achieved real performance is very close to the peak one.

Figure 3 shows a stage of Rayleigh-Taylor instability evolution at the same model time, "grown up" from the same initial state obtained by the "fast" and direct methods.

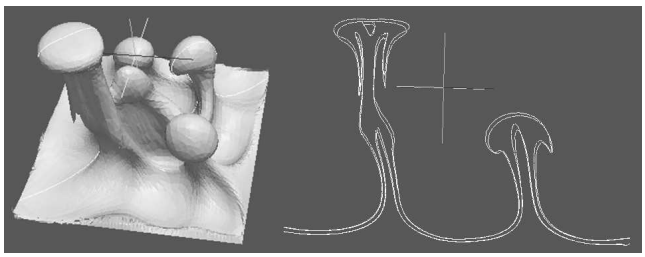


Figure 3: A stage of the model evolution. Left side shows the surface of unstable layer, obtained by the fast method. Right side shows two identical cuts of unstable layer, one was obtained by the direct method, another — by the fast algorithm. Computational grid size is $100 \times 100 \times 100$ **nodes.**

The direct algorithm is more precise to calculate velocity field directly on the surfaces. The fast algorithm provides $\mathbf{v}$ on the uniform grid, so it is necessary to interpolate (linear interpolation was used) velocity field on surfaces to solve (4). Due to this we have some difference on the cuts (see figure 3 right)[3]. In table 1 the performance gain, achieved by the fast algorithm, is shown.

Since the direct algorithm calculates velocity filed $\mathbf{v}$ only on the boundaries (surfaces S_i), the execution time does not quadratically depend on the grid size. Sometimes it is necessary to get $\mathbf{v}$ on the regular grid (the same as density

[2]Depending on the graphics accelerator

[3]This difference is quite acceptable for desired application — simulation of salt diapirism

Table 1: Calculation time on Tesla® M2090 (all evolution)

Grid size	Time (min)		Speedup
	direct (CUDA)	fast (cuFFT)	
75 × 75 × 75	11.4	1.05	**10.8x**
100 × 100 × 100	27.0	2.84	**9.5x**
150 × 150 × 150	144	12.5	**11.5x**

one), and in this case fast algorithm is much more efficient. In the table 2 the times of velocity field **v** calculation on uniform grid are shown.

Table 2: Calculation time on Tesla M2090 (v calculating on uniform grid)

Grid size	Time (sec)		Speedup
	direct (CUDA)	fast (cuFFT)	
64 × 64 × 64	18	1.49	**12.0x**
128 × 128 × 128	1133	2.61	**434.1x**
256 × 256 × 128	18157	29.04	**625.2x**

We also did some tests with famous multithreaded FFTW library on single CPU Intel® Core™ i5-4440, and compared it with direct implementation on single GPU Tesla M2090. Figure 4 clearly shows the advantages of the proposed fast method for calculating **v**. Note that the fast method works on single CPU using FFTW library much faster than the direct algorithm on GPU.

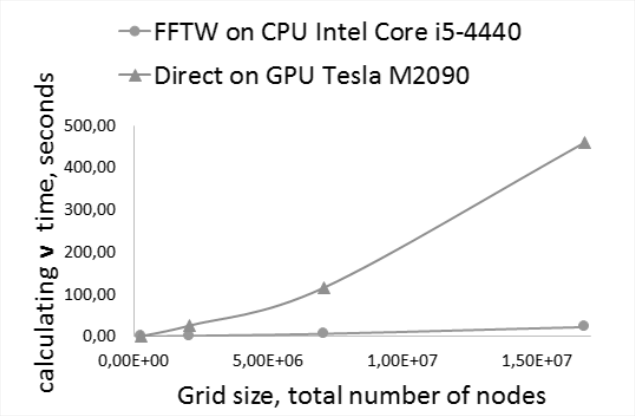


Figure 4: Comparison of calculation speed of fast implementation on CPU and direct one on single GPU depending on the grid size.

5. CONCLUSION

The method for fast numerical simulation of 3D gravitational creeping flow of uniform viscous incompressible Newtonian fluid with piecewise constant density in a half-space bounded by a free surface have been proposed, implemented and tested.

This method is extremely efficient on grids with a large number of nodes (10^6 and more) due to low computational complexity of the algorithm. As the result, valuable acceleration of numerical solution of the given mathematical physics problem, is achieved. The considered problem has great practical importance as oil and gas deposits are often located at salt domes.

Note that evaluating (6) via cyclic convolution (7) is exact, i. e. direct and fast algorithms provide precisely the same results neglecting rounding errors.

6. ACKNOWLEDGMENTS

The work was conducted as a part of the Program of Basic Scientific Research VIII.73.2 SB RAS and supported by the Russian Foundation for Basic Research (Grant no. 16-35-00443 mol-a).

The authors also wish to thank Valery N. Martynov[4] for the idea of applying the convolution theorem to optimize the direct algorithm.

7. REFERENCES

[1] T. V. Abramov. Massively parallel rayleigh-taylor instability simulation using analytical expression of greens function of the corresponding boundary value problem. *Computational technologies*, 20(4):3–16, 2015.

[2] T. V. Abramov. Fast numerical solution of boundary value problems with known greens function using cyclic convolution. *Computational technologies*, 21(2):3–11, 2016.

[3] M. R. Hudec and M. P. A. Jackson. Terra infirma: Understanding salt tectonics. *Earth Science Reviews*, 82(1):1–28, 2007.

[4] A. T. Ismail-Zadeh, I. A. Tsepelev, C. Talbot, and P. Oster. Three-dimensional modeling of salt diapirism: A numerical approach and algorithm of parallel calculations. *Comput. Seism. Geodyn. Amer. Geophys. Union*, (6):33–41, 2004.

[5] B. V. Lunev. Isostasy as the dynamic equilibrium of a viscous fluid. *Doklady AN SSSR*, 290(6):72–76, 1986.

[6] B. V. Lunev and V. V. Lapkovskii. Fast numerical simulation of salt tectonics: possibility of the operational application in geological practice. *Fiz. mezomekh.*, 12(1):63–74, 2009.

[7] B. V. Lunev and V. V. Lapkovskii. Mechanism of development of inversion folding in the subsalt. *Izvestiya, Physics of the Solid Earth*, 50(1):57–63, 2014.

[8] H. Nussbaumer. *Fast Fourier Transform and Convolution Algorithms. Second Corrected and Updated Edition.* Springer-Verlag Berlin and Heidelberg GmbH & Co. KG, Berlin, 1990.

[9] C. J. Talbot and M. P. A. Jackson. Salt tectonics. *Scientific American*, 257(2):70–79, 1987.

[4]Institute of Computational Mathematics and Mathematical Geophysics SB RAS

Theoretical Study of Replication in Desktop Grid Computing: Minimizing the Mean Cost

Ilia Chernov
Institute of Applied Math Research, Kareliam Research Centre of RAS
185910 Pushkinskaya 11
Petrozavodsk, Russia
chernov@krc.karelia.ru

ABSTRACT

We consider a model of computing process with independently produced results. Enterprise desktop grid is kept in mind as a computing tool. All nodes are equal. Possibility of producing wrong results is taken into account. We assume that a priori distribution of possible answers and probabilities of producing one answer while another is correct are known. In case a wrong result is accepted, some penalty is added to the computation cost. m-first voting replication scheme is used to minimize the overall average expenses. A task is replicated until a given number of identical answers are obtained. The problem is to choose the optimal quorums which can depend on answers. We consider the most general model and show how to solve the optimization problem. A few simple and asymptotic cases are studied. The main conclusion is that optimal quorums are quite stable with respect to penalties, so there is no need to know their exact values. Also we consider two groups of computing nodes and show on an example that well-chosen replication scheme on weaker computers can be better than using faster ones.

Categories and Subject Descriptors

C.4 [**Computer Systems Organization**]: Performance of systems—*Fault tolerance*; C.2.4 [**Computer Systems Organization**]: Performance of systems—*Distributed systems*

General Terms

Theory

Keywords

desktop grid, replication, reliability, volunteer computing, optimal quorum

1. INTRODUCTION

Desktop grids have become a cheap and rather powerful tool for solving various problems from different branches of science. Enterprise desktop grids (EDG, see, e.g., [1]) gather desktop computers, servers, and other resources from one or a few institutions to solve multiple computing tasks connecting via LAN or Internet. EDG does not suffer from malicious actions and unpredictable switch-offs; However, wrong results can be returned due to a number of reasons. This can be malfunction of hardware, corruption of data, algorithmic errors, wrong results produced by the correct algorithm. An example of the latter case is possible convergence of a descent method to false minima.

Most errors can be revealed by replication: solving the same problem a few times on different computers. We use the quorum approach, also called m-first voting: the result task is replicated until ν identical results are obtained to be accepted as the truth. Of course, the quorum ν can depend on the answer, i.e., some answers can be checked more carefully. Note that this approach differs from the majority voting where the answer received most times from N replicas is believed to be true: in this case redundancy is fixed, while in the m-first voting number of replicas can grow quite significantly; on the other hand, majority voting can fail to produce a result and may lack reliability.

Choosing the quorum values is not an easy problem. If too high, much resources would do useless work, though underestimation is too risky. We minimize the average cost of computation per a task as expected expenses on solving a task a few times with expected pelaties paid in case a wrong answer has been accepted. To the best of our knowledge such approach of minimizing the total mean cost including spent time and possible penalties has not been considered so far.

Let us consider virtual screening [2] as an example. Software is able to evaluate energy of binding between a small molecule called ligand and a larger protein molecule. Calculation can be performed using different parameters, precision and other, so false solutions sometimes appear. Algorithms use random numbers, therefore, a few tries even on similar computers are able to reveal the mistake. In case such wrong answer is accepted, the substance is recommended for biochemical test in a laboratory, which is rather costly; this cost together with reputation losses, pointless use of computing resources, etc, forms rather high, compared to the cost of a single computation, penalty.

A drawback of such approach is lack of precise information about the penalty values. It is by no means easy to estimate losses in case of producing a wrong answer in terms of cost of an individual task, especially in heterogeneous Desktop grids. However, as we show, rough estimation is enough:

ICAIT '16, Oct. 6 – 8, 2016, Aizu-Wakamatsu, Japan.

optimal quorums are rather stable with respect to penalty values, at least in practically important case of low risk.

Beside solving the problem of optimal replication given risk levels for individual calculation and penalty threats, one can consider another problem: to choose penalties that force the desired replication (at least given or higher) with minimal possible average cost.

Although we focus on EDG, the presented approach can be applied also to volunteer computing systems, because the concept of probability of the correct answer can be used, with some restrictions, to counteracting saboteurs also.

The rest of the article is organized as follows. We review the related work, then propose the model, as general as possible, and derive formulae for probabilities and the mean cost that can be used in practice. Also we give hints on how to solve the optimization problem. We consider a few simple cases: one with an absolutely reliable answer and a recognition problem. Here we are able to solve the optimization problem analytically. We define the concept of *critical penalties* that force change of quorums; typically they are quite rare, so there is no need to know the penalty exactly. This fact is important for practical use of our approach. We develop this idea by considering asymptotic analysis of the model in case of low risk. Also we show that quicker (or cheaper) but less reliable computers can over-compete slower (more expensive) at the same level of risk (and higher redundancy). In the conclusion plans for future are described.

2. RELATED WORK

By replication in this article we mean always task replication. Replication has been used in desktop grid computing since its creation: BOINC middleware [3] supports redundant computing to identify wrong results. The survey [4] reviews fault-tolerance techniques, including job replication; though other methods discussed there hardly can help to reveal mistakes: they reduce losses due to unexpected switch-offs, unfinished tasks, etc.

Beside increasing reliability of the system, replication is often used for optimizing productivity of the computing system using a few metrics, when it is important to complete a task as soon as possible, even paying the cost of redundancy. One of the metrics is makespan: time for completing all tasks. For example, [5] considers a typical EDG computing situation with rather low amount of tasks; so near the end of computing process there are free resources while each error or switch-off drastically slows down the whole process. Duplication of tasks improves the makespan.

In [6, 7] replication is studied theoretically from the point of view of improving productivity of multicomputer system. The authors of [8] consider how to reduce loss of computation power due to replication by choosing optimal replication according to nodes' reliability. In [9] replication is used to reduce risk of violating deadlines in case of unreliable computers. Article [10] considers the problem from a different point: sometimes it is necessary to run replicas on absolutely identical computers due to strong dependence of results on precision and other technical details of computing process.

Much attention has been payed to replication as one of the methods ([11] is a review of them) for revealing malicious actions (sabotage) in volunteer computing systems.

The work [12] studies errors is Internet computing grids. It provides some statistically obtained estimates: about one third of hosts return a wrong result at least once; the mean error rate is about 0.002; majority voting is shown to be able to reduce risk to $2 \cdot 10^{-4}$. Also errors from hosts showed no correlation: this allows to extend our approach to Internet computing also.

3. THE MATHEMATICAL MODEL

3.1 Assumptions

Let x_i, $1 \le i \le I$ be the set of possible answers to some problem. The nature of x_i does not matter: this can be real or integer numbers, matrices, vectors, texts, boolean values, etc. However, some knowledge is usually available about distribution of x_i: some are more expected than others; let us denote the *a priori* distribution of x_i by α_i.

Answers are produced by some computing system (we keep in mind an Enterprise desktop grid). What is important for the model is only the fact that different answers are produced independently.

As we have noted above, there is often probability of mistake: producing a wrong answer in desktop grid computing. The reasons can be different, as we have noted above. The probability of error can, in general, depend on what answer is indeed correct. So let us denote the probability of getting the answer x_i while x_j is true by p_{ij}. In the absolutely reliable system p_{ij} is the Kronecker delta-symbol.

In case of accepting a wrong answer we suffer some penalty, either by some kind of direct fine, or loosing reputation, spending funds on needless examining the object in lab, doing much needless work, etc. Let us denote the penalty paid in case of accepting x_i while x_j is correct by F_{ij}.

All computing nodes are identical. Let the cost of producing an answer s_i is C_i. In the end of the article we consider two groups of computing nodes of different reliability and efficiency.

So, the *a priori* distribution α_i of the answers, conditional probabilities p_{ij} of getting an answer x_i while another answer x_j is correct, penalty values F_{ij}, and costs C_i are known. As we have already noted and will prove in the sequel, in practive it is sufficient to know penalties with very low precision. The order of magnitude of penalties (with cost of an average task as the cost unit) can be expected to be known in most cases.

Risk of getting an error can be estimated if some statistics of previous calculations is available. In the simplest case there are only the error probability and the probability of the correct answer. They are usually known for an algorithm. More subtle estimations are often available.

A priori distribution of possible answers can be also available if there is any statistics. To use virtual screening as an example, there are data bases that give binding energy of ligands and proteins; they allow to estimate probability distribution of different energies. Molecules have different chances to show high predicted binding affinity. These chances are expected to be higher for molecules close by topology to a known ligand [14]. In contrast, molecules with very large number of atoms are less likely to bind well [15]. So, knowledge about chemical structure of molecules can provide estimations of probabilities of the binding energy, i.e., of the possible answers.

History of the search allows to improve these estimations. Some ligands are similar, so results for one serve as source of estimations for another. The same is true for costs C_i of the tasks. In the simplest case they all equal 1; if any additional

information is available, C_i can be estimated more precisely.

Again, results are rather stable with respect to distribution of the answers. What is practically important, is whether one answer is much more rare than the other, or if ratios of probabilities of answers differ much from that of penalties if these answer were wrong. Such facts can be expected to be known, at least in case of simple structure of the answer set, like yes/no or good/normal/bad.

Considering the most general construction is useful also for another reason. Understanding the mechanism helps to organize calculation in an effective way even without precise knowledge about the input data and, then, without exact solution.

3.2 Replication

To protect ourselves from additional losses we need to minimize our total cost. In this paper we analyze replication as a means for that. Too high redundancy does not decrease risk any more, but increases additional work linearly. However, low quorum means high risk and thus expected penalties are too large.

We believe an answer x_i to be correct if it arrives exactly ν_i times from computing nodes; note that the same task can produce other answers on other nodes: we accept the answer already obtained $\nu_i - 1$ times as soon as it is obtained once more, no matter how many times other answers arrived (provided that less than their quorums). In the simplest case of yes or no answers we can distinguish them checking one more carefully [13].

So the total cost of solving a task consists of solving it a few times with the same answer, possibly solving it more times with other answers, and the penalty F_{ij}. This cost is a random variable; we want to minimize its mean value.

To evaluate the mean we need to consider all possibilities of believing a given x_i while some x_j is the correct answer. The final obtained result must be, obviously, x_i: it stops the process completing the quorum. Before that we can have all x_m, in any order, but each can arrive at most $\nu_m - 1$ times (otherwise it would have been believed). The number of all possibilities equals $K_i = \prod_{m=1,m\neq i}^{I} \nu_m - 1$. Having a possibility with a non-negative number $k < K$, we can produce all the numbers $R_{k,m}$ of times an answer x_m has been obtained, $m \neq i$, using the following formulae: $M_{k,0} = k\nu_i$,

$$M_{k,m} = \left[\frac{M_{k,m-1}}{\nu_m}\right], \quad R_{k,m} = \mathrm{mod}(M_{k,m-1}, \nu_m).$$

Note that $R_{k,m}$ depend on i, though the notation hides this dependence. If a k is chosen, the probability that all x_m, $m \neq i$ were obtained exactly $R_{k,m}$ times in any order, the answer x_i was obtained exactly $\nu_i - 1$ times also arbitrarily distributed among other answers, and that it was obtained once more as the final answer is

$$P_{k,i,j} = \frac{\left(\nu_i - 1 + \sum_{m=1,m\neq i}^{I} R_{k,m}\right)!}{(\nu_i - 1)! \prod_{m=1,m\neq i}^{I} R_{k,m}!} \prod_{m=1,m\neq i}^{I} p_{mj}^{R_{k,m}} p_{ij}^{\nu_i}.$$

This is the conditional probability assuming that x_j is truly correct while x_i is believed to be correct, and k-th distribution of other answers has happened. The cost in this case consists of computational expenses for evaluating all rejected answers:

$$\sum_{m=1,m\neq i}^{I} R_{k,m} C_m,$$

expenses on evaluating the accepted task x_i with ν_i replicas: $C_i\nu_i$, and penalty F_{ij}. The sum

$$E_{i,j} = \sum_{k=0}^{\prod_{m=1,m\neq i}^{I} \nu_m - 1} P_{k,i,j} \cdot \Big(\sum_{m=1,m\neq i}^{I} R_{k,m} C_m + C_i\nu_i + F_{ij} \Big)$$

is the expected cost in case x_j is correct while x_i has been accepted. Summing them all up over $i = 1$ to I we get E_j: the expected cost in case x_j is the correct answer. Finally we need to take the *a priori* distribution of correct answers into account to get the expected cost

$$E = \sum_{j=1}^{I} \alpha_j E_j.$$

4. OPTIMIZATION

For given penalties F_{ij} the problem of finding the optimal quorums is to choose such ν_i that E has the minimal possible value. We can assume that $F_{i,i} = 0$ for all i or at least these values are much less compared to other F_{ij}: no penalties for the correct answer. Also error probability must be low enough, otherwise computing has no sense at all: $p_{ii} > 0.5$. Under these assumptions it is clear that if all $\nu_i > \bar{\nu}$ for some $\bar{\nu}$, then the expected penalty is low enough, so that asymptotically $E \sim \sum_{j=1}^{I} \alpha_j C_j \nu_j$ and therefore grows at least linearly with respect to ν. Therefore, choosing high enough ν and then decreasing it, we quickly find an upper bound for all quorums; then further individual reducing of ν_i provides lower bounds.

4.1 Simple cases

In the simplest case there are only two possible answers. Then $I = 2$, $R_{k,m} = k$, $P_{k,i,j} = \binom{\nu_i - 1 + k}{k} p_{3-i,j}^{k} p_{ij}^{\nu_i}$, and

$$E = \sum_{i,j=1}^{2} \alpha_j \sum_{k=0}^{\nu_w - 1} \binom{\nu_i - 1 + k}{k} p_{w,j}^{k} p_{ij}^{\nu_i} \Big(kC_w + C_i\nu_i + F_{ij} \Big)$$

(here $w = 3 - i$). Further simplification is assuming that there is just the correct and the wrong answer, so p_{ij} and F_{ij} are symmetrical 2×2 matrices, $p_{ii} = q$, $F_{ii} = 0$, $\nu_i = \nu$, $C_i = C$. Then α_j do not matter and

$$E = \sum_{i=1}^{2} \sum_{k=0}^{\nu-1} \binom{\nu - 1 + k}{k} p_{3-i,1}^{k} p_{i,1}^{\nu} \Big(kC + C\nu + F_{i1} \Big).$$

Another simple case appears when an answer is absolutely reliable, so that $p_{jj} = 1$ for some fixed j; then, obviously, $p_{ij} = 0$ for all $i \neq j$.

Let us consider the simplest case of two possible answers of equal value with probabilities $q > 0.5$ and $p = 1 - q$ and penalty F in case of the wrong answer has been accepted. Then quorum ν is not optimal if

$$\Delta E = E(\nu + 1) - E(\nu) = \Delta E_0 - AF < 0.$$

Here A is some quantity dependent on p and ν but not on F, while ΔE_0 is ΔE in case of no penalty ($F = 0$) and therefore

is positive. As for A, it is equal to

$$A = p^\nu q^\nu \binom{2\nu-1}{\nu-1}(1-2p).$$

We omit the proof here.

The critical penalty F such that $\Delta E \approx 0$ grows quickly as $p \to 0.5$ and much more quickly as $p \to 0$. For example, in case of $p = 10^{-3}$ we have $A \approx 1.5 \cdot 10^{-8}$ and $\Delta E_0 \approx 1$, so that the critical penalty would be $F^* \approx 0.66 \cdot 10^8$. For $p \approx 10^{-2}$ we have $F^* \approx 0.66 \cdot 10^5$. This asymptotic analysis is applied to the more general case in the next section.

4.2 Asymptotic analysis

Let us assume that probability of an error is negligibly low, so that only threat of penalty justifies taking it into account. Thus we neglect p_{ij} for $i \neq j$ and $1 - p_{ii}$ compared to p_{ii} and 1. Then either the correct answer x_j is obtained ν_j times in a row (other possibilities are too unlikely to consider), or a wrong x_i is accepted with the correct x_j seen any number of times from 0 to ν_{i-1}: all these cases are approximately equally probable. Then $P_{k,i,j}$ becomes simpler: x_i is accepted, another x_j is correct, it is obtained k times with $p_{ij} \approx 1$, $R_{k,m} = k$, m does not vary from 1 to I, instead $m = j$. So, finally:

$$P_{k,i,j} = \binom{\nu_i - 1 + k}{k} p_{ij}^{\nu_i}.$$

If the accepted answer is correct, $P_{k,j,j} \approx 1$, k can be only 0: receiving other answers is highly unlikely. Then

$$E = \sum_{j=1}^{I} \alpha_j \Big(C_j \nu_j + F_{jj} \Big) +$$
$$+ \sum_{j=1}^{I} \alpha_j \sum_{i=1, i\neq j}^{I} \sum_{k=0}^{\nu_j - 1} \binom{\nu_i - 1 + k}{k} p_{ij}^{\nu_i} \Big(k C_j + C_i \nu_i + F_{ij} \Big).$$

We are interested in increment of this cost when a quorum is changed:

$$\Delta E_J = E(\nu_J + 1) - E(\nu_J) =$$
$$\alpha_J C_J + \alpha_J \sum_{i=1, i\neq J}^{I} \binom{\nu_i - 1 + \nu_J}{\nu_J} p_{iJ}^{\nu_i} \Big(\nu_J C_J + C_i \nu_i + F_{iJ} \Big) -$$
$$\sum_{j=1, j\neq J}^{I} \alpha_j \sum_{k=0}^{\nu_j - 1} \binom{\nu_J - 1 + k}{k} p_{Jj}^{\nu_J} \Big(k C_j + C_J \nu_J + F_{Jj} \Big).$$

Now we see that E grows linearly with gradient $\alpha_J C_J$ with respect to ν_J provided that all ν_i are high enough. This means that rare valuable answers, with low probability α_J can be examined carefully without significant losses: ν_J can be taken much more than the optimal value.

Another fact is that not penalties but quantities $p_{ij}^{\nu_i} F_{ij}$ matter. As p_{ij} are low, changes of ν_i modify such quantities very significantly provided that penalties are large compared to costs C_i. Let us say that a quorum $\nu_J > 1$ is equilibrium if $\Delta E_J \geq 0$ while $\Delta E_{J-1} \leq 0$. If a quorum ν_J is equilibrium, it is stable with respect to small changes of the penalty $F_{J,i}$: significant changes of $F_{J,i}$ are such that $F_{J,i} p_{Ji}^{\nu_J}$ remain approximately the same. This means, again, that in case of low risk and high penalties we do not need to know precise values of penalties for accepting a wrong answer: it suffices to know the order of magnitude with precision about p^{-1} where p is the maximal error probability.

A consequence of this is the notion of critical penalties. Let us consider the problem of choosing penalties F_{ij} such that the desired quorums (and thus the desired probability of an error) were optimal. This is a linear optimization problem with an utility function E and constraints of the form $\Delta E_{J-1} \leq 0$. As the utility function has positive coefficients and constraints of the form $F_{ij} \geq 0$ are valid, the problem has a solution provided that at least one admissible point exists. An admissible point is such set of penalties that justifies passing from a quorum ν_{i-1} to ν_i. It is possible that there is no such point, for example if all $p_{Jj} = 0$. Then no penalty is able to make any replication pay. However, if the third term in the expression for ΔE_J is not zero and quorums are high enough, an admissible point always exists and so does a solution to the linear optimization problem for the optimal penalties. These penalties are called critical. If real penalties are close to these ones, even slight difference in computing cost is able to change optimal quorums.

Let us consider another asymptotic case. Assume that the number of possible answers I is so high that probability of receiving a wrong answer twice is negligibly small. Then replication $\nu = 2$ is always sufficient. However, multiple wrong answer are able to arrive prior to the second correct answer, so that amount of work can be high. Let us assume that error risk is the same for all wrong answers, probability of the correct answer is q, $p = 1 - q$ is risk of getting a wrong answer, and cost of each answer is unit. Then we can get from $0 \leq T \leq I - 1$ wrong answers, at most once each, exactly one correct answer at some position and, finally, the second correct answer. Amount of work for each case is $T + 2$, so the mean cost then is

$$E_2 = \sum_{T=0}^{I-1} (T+2)(T+1) p^T q^2 = \frac{2}{q}.$$

The factor $T + 1$ counts number of positions where the first correct answer can be among T wrong ones. So, for reliable computers expected cost is near 2 while for bad ones it can be very high.

We need to compare it with the no-replication case, where the first obtained answer is accepted with penalty F in case of an error. The expected cost is

$$E_1 = 1 + pF.$$

So the critical penalty is

$$F^* = \frac{2 - q}{pq}$$

Again for reliable computers (with small p) critical penalties are of order p^{-1}. Though the minimal possible value is only ≈ 5.83 at $q = 2 - \sqrt{2} \approx 0.586$.

4.3 Desired risk level

Instead of choosing the desired quorums and looking for the critical penalties, one can consider another problem. Choose the desired risk level ρ and demand that the total probability of accepting a wrong answer is at most ρ. This probability is, obviously, E in the special case of $C_i = 0$, $F_{ii} = 0$, $F_{ij} = 1$ for $i \neq j$. Denote this value P. Then the optimization problem has one constraint and looks as

$$E \to \min, \quad P \leq \rho.$$

This problem has a solution provided that $p_{jj} > p_{ij}$ for all i and j: probability of accepting a particular wrong answer would reduce more quickly than that of the right answer. Thus very high quorums make the risk arbitrary low, so that quorums that provide $P \leq \rho$ exist. Then note that E grows with respect to each ν_j provided that all ν_i are high enough and that it grows with respect to each F_{ij}. This fact, together with positiveness of ν_i and $F_{i,j}$, $i \neq j$, guarantees existence of a solution to the optimization problem.

4.4 Different nodes

Assume that we have two groups of computing nodes of different reliability p_{ij} and costs C_i. If we can choose what group to use, we need to compare average overall cost for these two groups under optimal replication. Let us assume that quorums are high enough so that effective expected penalties $p_{ij}^{\nu_i} F_{ij}$ are comparable with costs C_i. Then the quantities $W_{ij} = C_i \nu_i + p_{ij}^{\nu_i} F_{ij}$ can be used to decide which group is better. As an example, let us consider one group with $C_i = 2$ and $p_{ij} = 0.01$ and another group with $C_i = 1$, $p_{ij} = 0.02$. The optimal quorums are $\nu_i = 5$ and $\nu_i = 6$. Then the first group has $W_{ij}^1 = 10 + 0.01^5 F_{ij}$, while the other's $W_{ij}^2 = 6 + 0.02^6 F_{ij} = 6 + 0.64 \cdot 0.01^5 F_{ij} < W_{ij}^1$. This simple example shows that well-chosen replication is able to cope with lower reliability.

5. CONCLUSIONS AND FUTURE WORK

We have considered a model of grid computing with risk of error and threat of penalty and the problem of minimizing the total mean cost of computation consisting of cost of redundant solution of tasks and estimated penalties. The main results are:

- The model, as general as possible; a way to solve the optimization problem.
- The notion of critical penalties, their dependence on risk probabilities.
- Stability of the optimal solution with respect to penalty values, practically important.
- Low harm of overestimated quorums.
- Possibility of improving poor computing by replication.

In future we plan to consider productivity of the Desktop grid together with optimizing the mean cost. Optimal scheduling of replicas is one interesting question. For example, for quorum 2 sometimes it is better to send two replicas hoping for identical answers, while in other cases three replicas should be sent at once so that decision be taken during one time unit. Fighting with saboteurs is even able to force one-by-one scheduling of replicas. We only slightly have considered heterogeneity of computing nodes; though another question is optimal scheduling of optimal number of replicas among heterogeneous computing nodes. The third question to be studied is considering productivity in terms of award that is able to reduce the total average cost. This approach seems to be almost unstudied, so far.

6. ACKNOWLEDGMENTS

The work has been supported by Russian Foundation for Basic Research: projects 16-07-00622 and 15-29-07974.

7. REFERENCES

[1] E. Ivashko, Enterprise desktop grids, in: Proceedings of the II International Conference BOINC-based High Performance Computing: Fundamental Research and Development (BOINC:FAST 2015), Vol. Vol-1502, CEUR Workshop proceedings, 2015, pp. 16–21.

[2] J. Kovács, P. Kacsuk, A. Lomaka, Using a private desktop grid system for accelerating drug discovery, Future Generation Computer Systems 27 (2011) 657–666.

[3] D. P. Anderson, BOINC: A system for public-resource computing and storage, in: Grid Computing, 2004. Proceedings. V IEEE/ACM International Workshop on, 2004, pp. 4–10.

[4] S. Siva Sathya, K. Syam Babu, Survey of fault tolerant techniques for grid, Computer Science Review 4 (2) (2010) 101–120.

[5] D. Kondo, A. A. Chien, H. Casanova, Scheduling Task Parallel Applications for Rapid Turnaround on Enterprise Desktop Grids, Journal of Grid Computing 5 (4) (2007) 379–405.

[6] E. C. Xavier, R. R. S. Peixoto, J. L. M. da Silveira, Scheduling with task replication on desktop grids: theoretical and experimental analysis, Journal of Combinatorial Optimization 30 (3) (2015) 520–544.

[7] A. Benoit, Y. Robert, A. L. Rosenberg, F. Vivien, Static strategies for worksharing with unrecoverable interruptions, Theory Comput Syst 53 (2013) 386–423.

[8] M. K. Khan et al,A Group Based Replication Mechanism to Reduce the Wastage of Processing Cycles in Volunteer Computing, Wireless Personal Communications 76 (3) (2014) 591–601.

[9] E. M. Heien, D. P. Anderson, K. Hagihara, Computing Low Latency Batches with Unreliable Workers in Volunteer Computing Environments, Journal of Grid Computing 7 (4) (2009) 501–518.

[10] M. Taufer, D. Anderson, P. Cicotti, C. L. Brooks III, Homogeneous redundancy: a technique to ensure integrity of molecular simulation results using public computing, in: Parallel and Distributed Processing Symposium, 2005. Proceedings. 2005, pp. 119a–119a.

[11] L. F. G. Sarmenta, Sabotage-tolerance mechanisms for volunteer computing systems (2002).

[12] D. Kondo et al, Characterizing result errors in internet desktop grids, in: A.-M. Kermarrec, L. Bougé, T. Priol (Eds.), Parallel processing. Proceedings of XIII International Euro-Par Conference, Vol. 4641 of Lecture Notes in Computer science, Springer, 2007, pp. 361–371.

[13] I. Chernov, N. Nikitina, Virtual screening in a desktop grid: Replication and the optimal quorum, in: Parallel Computing Technologies: 13 International Conference, Springer, 2015, pp. 258–267.

[14] D.E. Patterson et al., Neighborhood behavior: a useful concept for validation of "molecular diversity" descriptors, Journal of Medicinal Chemistry 39 (1996) 3049–3059.

[15] M.M. Hann, A.R. Leach, G. Harper, Molecular complexity and its impact on the probability of finding leads for drug discovery, Journal of Chemical Information and Computer Sciences 41 (2001) 856–864.

AUTHOR INDEX

AUTHOR INDEX

Proceedings of the 2nd International Conference on
Applications in Information Technology (ICAIT-2016)

THE UNIVERSITY OF AIZU
Tsuruga, Ikki-machi, Aizu-Wakamatsu City
Fukushima, 965-8580 Japan
phone: +81-242-37-2500 fax: +81-242-37-2528

THE UNIVERSITY OF AIZU

ISBN 978-4-900721-04-3