Smart Innovation, Systems and Technologies 217

Lakhmi C. Jain
Margarita N. Favorskaya
Ilia S. Nikitin
Dmitry L. Reviznikov *Editors*

Applied Mathematics and Computational Mechanics for Smart Applications

Proceedings of AMMAI 2020

KES International

Springer

Smart Innovation, Systems and Technologies

Volume 217

The Smart Innovation, Systems and Technologies book series encompasses the topics of knowledge, intelligence, innovation and sustainability. The aim of the series is to make available a platform for the publication of books on all aspects of single and multi-disciplinary research on these themes in order to make the latest results available in a readily-accessible form. Volumes on interdisciplinary research combining two or more of these areas is particularly sought.

The series covers systems and paradigms that employ knowledge and intelligence in a broad sense. Its scope is systems having embedded knowledge and intelligence, which may be applied to the solution of world problems in industry, the environment and the community. It also focusses on the knowledge-transfer methodologies and innovation strategies employed to make this happen effectively. The combination of intelligent systems tools and a broad range of applications introduces a need for a synergy of disciplines from science, technology, business and the humanities. The series will include conference proceedings, edited collections, monographs, handbooks, reference books, and other relevant types of book in areas of science and technology where smart systems and technologies can offer innovative solutions.

High quality content is an essential feature for all book proposals accepted for the series. It is expected that editors of all accepted volumes will ensure that contributions are subjected to an appropriate level of reviewing process and adhere to KES quality principles.

Indexed by SCOPUS, EI Compendex, INSPEC, WTI Frankfurt eG, zbMATH, Japanese Science and Technology Agency (JST), SCImago, DBLP.

All books published in the series are submitted for consideration in Web of Science.

More information about this series at http://www.springer.com/series/8767

Lakhmi C. Jain · Margarita N. Favorskaya ·
Ilia S. Nikitin · Dmitry L. Reviznikov
Editors

Applied Mathematics and Computational Mechanics for Smart Applications

Proceedings of AMMAI 2020

Springer

Editors
Lakhmi C. Jain
KES International
Shoreham-by-sea, UK

Liverpool Hope University
Liverpool, UK

University of Technology Sydney
Sydney, Australia

Ilia S. Nikitin
Institute of Computer Aided Design (ICAD)
Russian Academy of Sciences (RAS)
Moscow, Russia

Margarita N. Favorskaya
Department of Informatics and Computer Techniques
Institute of Informatics and Telecommunications
Reshetnev Siberian State University of Science and Technology
Krasnoyarsk, Russia

Dmitry L. Reviznikov
Moscow Aviation Institute
National Research University
Moscow, Russia

ISSN 2190-3018 ISSN 2190-3026 (electronic)
Smart Innovation, Systems and Technologies
ISBN 978-981-33-4825-7 ISBN 978-981-33-4826-4 (eBook)
https://doi.org/10.1007/978-981-33-4826-4

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Preface

This research book presents the selected papers reported at the 13th International Conference on Applied Mathematics and Mechanics in the Aerospace Industry (AMMAI 2020), which was held during 6–13 September 2020. The contributions include the modern numerical methods and mathematical models for solving problems of the computational mechanics, dynamic system simulation and optimization, information technologies, and artificial intelligence. Part I "Computational Fluid Dynamics" involves Chaps. 2–5, Part II "Numerical Simulation of Plasma and Multiphase Flows" contains Chaps. 6–11, Part III "Computational Solid Mechanics" includes Chaps. 12–14, Part IV "Numerical Study of Dynamic Systems" consists in Chaps. 15–20, and Part V "Information Technologies" contains Chaps. 21–24.

Each chapter was reviewed by two experts in the field, and the revised chapters were checked by the editors of the book.

We wish to express our gratitude to the authors and reviewers for their contributions. The assistance provided by Springer-Verlag is acknowledged.

Krasnoyarsk, Russian Federation — Margarita N. Favorskaya
Sydney, Australia — Lakhmi C. Jain
Moscow, Russian Federation — Ilia S. Nikitin
Moscow, Russian Federation — Dmitry L. Reviznikov

Contents

About the Editors

Dr. Lakhmi C. Jain is with the University of Technology Sydney, Australia, and Liverpool Hope University, UK. He is Fellow of the Institution of Engineers Australia. Professor Jain founded the KES International for providing a professional community the opportunities for publications, knowledge exchange, cooperation, and teaming. Involving around 5000 researchers drawn from universities and companies worldwide, KES facilitates international cooperation and generates synergy in teaching and research. KES regularly provides networking opportunities for professional community through one of the largest conferences of its kind in the area of KES. www.kesinternational.org.

Dr. Margarita N. Favorskaya is Professor and Head of Department of Informatics and Computer Techniques at Reshetnev Siberian State University of Science and Technology, Russian Federation. Professor Favorskaya is a member of KES organization since 2010, the IPC member, and Chair of invited sessions of over 30 international conferences. She serves as Reviewer in international journals (*Neurocomputing*, *Knowledge Engineering and Soft Data Paradigms*, *Pattern Recognition Letters*, *Engineering Applications of Artificial Intelligence*), Associate Editor of *Intelligent Decision Technologies Journal*, *International Journal of Knowledge-Based and Intelligent Engineering Systems*, and *International Journal of Reasoning-Based Intelligent Systems*, Honorary Editor of the *International Journal of Knowledge Engineering and Soft Data Paradigms*, Reviewer, Guest Editor, and Book Editor (Springer). She is the author/co-author of 200 publications and 20 educational manuals in computer science/engineering. She co-authored/co-edited seven books for Springer recently. She supervised nine Ph.D. candidates and is presently supervising four Ph.D. students. Her main research interests are digital image and videos processing, remote sensing, pattern recognition, fractal image processing, artificial intelligence, smart systems design, and information technologies.

Dr. Ilia S. Nikitin is Professor and Director ICAD RAS, Professor at Moscow Aviation Institute (MAI), a member of the Russian National Committee on Theoretical and Applied Mechanics, expert RAS, expert RSF, expert Minobrnauki RF. He graduated from Moscow Institute of Physics and Technology. His scientific interests

are mathematical modeling, numerical methods in continuum mechanics, moving adaptive meshes, dynamics of elastoplastic media, fatigue fracture, durability of operation, and high-frequency loading. The main scientific results are the numerical methods for solving non-stationary problems of continuum mechanics on moving and adaptive grids, methods for calculating the stress state of elements of aircraft structures and assessing the durability for various fatigue failure modes, refined models of layered and block media with different sliding conditions at the contact boundaries, the problems of propagation, transformation and reflection of waves in such media, and models of sintering powder materials under thermomechanical and pulsed high-energy effects.

Dr. Dmitry L. Reviznikov is Professor of the Department of Numerical Mathematics and Programming at Moscow Aviation Institute (National Research University), Russian Federation. Professor Dmitry L Reviznikov is a member of the Russian National Committee on Heat and Mass Transfer and a member of the Scientific Council of International Centre for Heat and Mass Transfer (ICHMT). He is Reviewer in international journals (*International Journal of Heat and Mass Transfer*, *Computational Thermal Sciences*, *International Journal of Fluid Mechanics Research*). He supervised eight Ph.D. candidates and is presently supervising three Ph.D. students. Scientific interests include mathematical modeling, computational physics, heat and mass transfer, multiphase flows, nonlinear dynamics, and data analysis. He is the author of more than 100 scientific papers in Russian and international journals and 4 monographs. Scientific results include fundamental results in the fields of modeling of conjugated heat and mass transfer, supersonic heterogeneous flows, thermal erosion destruction of heat-shielding coatings, anomalous diffusion, numerical methods for fractional differential equations, nonlinear wave dynamics, and interval analysis.

Chapter 1
Applied Mathematics and Mechanics in Aerospace Industry

Margarita N. Favorskaya, Lakhmi C. Jain, Ilia S. Nikitin, and Dmitry L. Reviznikov

Abstract The chapter contains a brief description of chapters that contribute to the development and applications of computational methods and algorithms in different areas of gas, fluid, and plasma dynamics, solid mechanics, dynamic systems and optimal control, information technology. The first part presents the recent advances in computational fluid dynamics. The second part introduces numerical simulation of plasma and multiphase flows. The third part is devoted to computational solid mechanics, the fourth part provides a numerical study of dynamic systems and the fifth part focuses on information technology and artificial intelligence.

M. N. Favorskaya (✉)
Reshetnev Siberian State University of Science and Technology, Institute of Informatics and Telecommunications, 31, Krasnoyarsky Rabochy ave., Krasnoyarsk 660037, Russian Federation
e-mail: favorskaya@sibsau.ru

L. C. Jain
University of Technology Sydney, Ultimo, Australia
e-mail: jainlc2002@yahoo.co.uk; jainlakhmi@gmail.com

Liverpool Hope University, Liverpool, England, UK

KES International, Shoreham-by-Sea, UK

I. S. Nikitin
Institute of Computer Aided Design of the RAS, 19/18, Vtoraya Brestskaya ul., Moscow 123056, Russian Federation
e-mail: i_nikitin@list.ru

D. L. Reviznikov
Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, Moscow 125993, Russian Federation
e-mail: reviznikov@inbox.ru

Federal Research Centre "Information and Control" of the RAS, 44, Ul. Vavilova, Moscow 119333, Russian Federation

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_1

1.1 Introduction

This book presents the selected papers reported at the 13th International Conference on Applied Mathematics and Mechanics in the Aerospace Industry (AMMAI'2020), which was held during 6–13 September 2020. The book includes modern numerical methods and mathematical models for solving problems of computational mechanics, dynamic systems simulation and optimization, information technologies, and artificial intelligence. Part I "Computational Fluid Dynamics" involves Chaps. 2–5, Part II "Numerical Simulation of Plasma and Multiphase Flows" contains Chaps. 6–11, Part III "Computational Solid Mechanics" includes Chaps. 12–14, Part IV "Numerical Study of Dynamic Systems" consists in Chaps. 15–20, Part V "Information Technologies" contains Chaps. 21–24.

1.2 Chapters in the Book

Part I presents the recent advances in computational fluid dynamics and includes 4 chapters.

Chapter 2 reports a numerical study of the flight of large bodies in the Earth's atmosphere [1]. Based on the model of a single body (no fragmentation), the authors determine the kinematic and physical characteristics necessary for a meteoroid to ascend in the atmosphere after its initial descend. It was found out that the key parameter for the possibility of such ascend is the angle of entry into the atmosphere. Authors compute the critical angles for a range of control parameters, i.e., the ballistic coefficient and the lift-to-drag ratio. The obtained results explain certain effects of the Tunguska event that took place in 1908.

Chapter 3 presents the results of numerical simulation of a hypersonic flow of a viscous heat-conducting gas near the landing module in the Martian atmosphere. The conservative numerical method of flux [2, 3] is used to solve the problem. Special attention is paid to the study of the structure of non-stationary flow on the side and bottom surfaces of the module. The results of the numerical simulation show that a developed unsteady vortex flow is realized on the lateral and bottom surfaces of the descent module, which affects the aerodynamic characteristics. It is important to note that the temperature near the lateral and bottom surfaces of the module can reach large values. These features of the flow must be taken into account when designing new aerospace vehicles.

Chapter 4 introduces the results of a comparison of two different numerical approaches for solving the problem of spot collapse: SMIF method and CABARET method [4, 5]. Several test tasks are considered and the results are compared with theoretical, experimental data and calculations of other authors.

Chapter 5 reports a numerical method for modeling the Taylor vortex flows [6]. The periodic boundary conditions on the edges of the cylinder's part are implemented. The results of the simulation for various values of the periodicity sizes and

different initial data are presented. When studying the flow of viscous gas between the cylinders of different temperatures, the flow modes with the flat vortex structures and with the Taylor vortices as well as the three-dimensional flow corresponding to the combination of these two flow types were found.

Part II introduces a numerical simulation of plasma and multiphase flows and involves six chapters.

Chapter 6 is devoted to mathematical modeling of gas-dynamic flows with the phase transformations (condensation and evaporation) [7, 8]. The system of the Navier–Stokes equations is used to describe the flow parameters, and the system of moment equations is used to describe the parameters of a two-phase medium. A numerical algorithm for solving the general system of equations is constructed on the basis of the Godunov scheme with the approximation AUSM + for solving the Riemann problem. The developed numerical model was optimized and adapted for the case of pure argon condensation in a nozzle based on Hagena's semi-empirical theory. In numerical experiments, certain values of the parameters of the condensation model are determined, such as accommodation coefficient and the nucleation correction factor multiplier.

Chapter 7 conducts a physical and comprehensive numerical study of the generation of plasma bunches with high specific energy with the use of a plasma gun [9, 10]. The parameters of the plasma bunch upon exit from the plasma accelerator and during propagation in the ionosphere ($h > 200$ km) to considerable distances ($\approx$100 km) have been calculated. A special numerical algorithm is presented to study the impact of a rarefied high-velocity gas flow (~5×10^7 cm/s) on the surface of crystalline and amorphous solid bodies. Based on the results, the electron concentration and the scale of the ionized region that formed during the passage of a high-speed toroidal plasma bunch through the rarefied air were estimated.

Chapter 8 is dedicated to the numerical study of pulsating gaseous detonation wave propagation. The mathematical model is based on the Euler equations written for the multicomponent gas and supplemented by the detailed chemical reactions model to describe the combustion of the hydrogen-air mixture [11, 12]. The Petersen and Hanson kinetics is applied as the detailed chemical model. The numerical algorithm is based on the finite volume approach, essentially non-oscillatory scheme, AUSM numerical flux, and the Runge–Kutta method. The numerical investigation of pulsating detonation wave propagation with direct detonation initiation near the closed end of the channel is carried out. The peculiarities of high-frequency and high-amplitude pulsations modes are discussed.

Chapter 9 considers a multi-temperature code for a multicomponent gas-dynamic [13, 14]. The gas-dynamic part is the Godunov-type method based on the efficient approximate solution of the Riemann problem operating with all components of the homogeneous gas mixture. The method assumes the table equation of state, but the system of the hydrodynamic equations should be hyperbolic. This work contains the test of the method on a strong shock wave in hydrogen plasma, so-called the Shafranov's solution. By taking into account the radiation component, the chapter discusses the applicability of the two temperature model for the strong shock wave in

the hydrogen with the large temperatures behind a shock wave without consideration of the radiation at a considered short timescale.

Chapter 10 deals with the issues of contactless removal of space debris objects, the orbit of which is changed by a high-velocity ion beam injected from the service spacecraft moving in the immediate vicinity of the debris object [15, 16]. Authors consider the problems of controlling the angles of rotation of electric propulsion thrusters to implement changes in the thrust components of the electric propulsion system in the longitudinal and transverse directions required during the debris object transportation. Arrangement of thrusters is proposed taking into account the location of solar arrays and the difference in permissible angles of thruster deflection in different planes.

Chapter 11 considers the current state of work on flight models of pulsed plasma propulsion systems [17, 18]. It is shown that the primary application area for propulsion systems based on an ablative pulsed plasma thruster is the station-keeping of small spacecraft with the power of supply system of up to 100 W and with an active lifetime in range from 1 to 10 years in low Earth orbits with altitudes in range from 400 to 700 km. It is also shown that ablative pulsed plasma thrusters can be efficiently used to solve the problems of accurate attitude control and angular stabilization of spacecraft.

Part III is devoted to computational solid mechanics and contains three chapters.

Chapter 12 presents a multi-mode kinetic model of cyclic loading damage development to describe the fatigue fracture process development [19, 20]. To determine the coefficients of the kinetic equation of damage, the well-known criterion of multiaxial fatigue failure SWT based on the mechanism associated with the development of microcracks of normal detachment is used. A numerical method for calculating crack-like zones up to macrofracture is proposed. The model parameters are determined from the condition of matching the experimental and calculated fatigue curve for a specimen of a certain geometry at a given load amplitude and cycle asymmetry coefficient. Using the obtained values, the results of experiments on specimens of a different geometry and asymmetry coefficients were reproduced numerically and the model and calculation algorithm performance were confirmed.

Chapter 13 is devoted to numerical modeling of elastic impacts on artificial ice islands arising as a result of drill impacts while drilling from the island, earthquakes, and pressure of structures located on the island, as well as collisions of the ice island with drifting ice layers [21, 22]. To solve this problem numerically, authors use the grid-characteristic method with interpolation on regular rectangular and parallelepipedal meshes and unstructured triangular and tetrahedral ones. The process of propagation of elastic waves in the considered geological environment is simulated and the distribution of stresses and the stability of the ice island to destruction are studied.

Chapter 14 contains the study of the impact of teeth installation parameters on the stress state of the prosthesis under typical chewing loads [23, 24]. Two main parameters are investigated: the role of the dentition installation line and the role of tilt angle of teeth blocks. The simple 3D models were developed and used for the

calculations. The results of the calculation show a higher sensitivity of the lower prosthesis basis to vary the parameters compared to the upper prosthesis basis.

Part IV provides a numerical study of dynamic systems and includes 6 chapters.

Chapter 15 discusses the framework of the restricted three-body problem [25, 26]. A celestial-mechanical model of the steady-state Chandler wobble of the Earth pole is proposed. The contribution of the astronomical and geophysical disturbances to the observed Earth pole oscillations is discussed based on the processing of IERS observations of the Earth pole motion, NCEP/NCAR geophysical data of the atmospheric circulation, and NASA/JPL angular momentum of the ocean. The Earth pole oscillatory process that is in-phase with the lunar orbit precessional motion is studied, and the contribution of moving media to this process is discussed.

Chapter 16 reports two multi-agent algorithms for controlling one class of continuous deterministic systems: a hybrid multi-agent method of interpolation search and a multi-agent method based on the use of linear regulators of agent movement control [27, 28]. Detailed descriptions of the strategies of these methods are given and step-by-step algorithms for each multi-agent method are described. Two approaches to the search for optimal open-loop control are considered: when control is sought in relay form with a certain number of switches, and when control is sought in the form of expansion in a system of basis functions.

Chapter 17 describes a modification for the continuous-time particle filter algorithm [29, 30]. The developed modification that is based on the well-known strategy such as modeling trajectories to numerically solve stochastic differential equations provides the lack of overflow errors during the calculation of particle weights. To implement such an idea practically particle weights should be expressed in terms of logarithms with additional customization of exponents. The effectiveness of the modified algorithm is demonstrated when solving the tracking problem to find coordinates and velocities of an aircraft executing a maneuver in the horizontal plane.

Chapter 18 proposes a solution to the problem of determining the contribution of airport rating criteria for assessing the integral risk of modernization. The purpose of modernization is to increase the throughput of the Moscow aviation hub [31]. The method proposed to solve the problem makes it possible to obtain the alternatives' weights for incomplete pairwise comparisons matrices of large dimension, as well as, alternative estimates in interval form, which is illustrated by an example. This method differs from most existing methods for solving the problem of incomplete pairwise comparisons by the ability to process incomplete pairwise comparisons matrices without restoring missing data [32]. It can be applied to solve other decision problems, where most of the known methods based on the pairwise comparisons method are not applicable.

Chapter 19 introduces the adaptive interpolation algorithm for systems with interval parameters [33, 34] and approaches directed to reducing the curse of dimensionality. The main assumption on which these approaches are based is that not all interval parameters make a significant contribution to the solution. The use of tensor train decomposition and sparse grids make it possible to take into account these features and expand the scope of the algorithm for the case of a large number of

interval parameters. The effectiveness of the considered approaches is confirmed on several model problems.

Chapter 20 discusses the spectral form of mathematical description for the representation of iterated Stratonovich stochastic integrals of an arbitrary multiplicity [35, 36]. Some invariant relations for expansion coefficients and iterated Stratonovich stochastic integrals are obtained which can reduce the computational cost. The spectral characteristics may be defined with respect to an arbitrary complete orthonormal system for the representation and modeling. For expansion coefficients, the tensor representation is formulated.

Part V focuses on information technology and artificial intelligence and includes 4 chapters.

Chapter 21 presents the fractal oriented approach [37, 38] for the analysis and design of distributed algorithms. Its aims are to represent the distributed algorithm as an "elastic object" that transforms dynamically at runtime. The use of the container-component model provides the following advantages: the ability to select automatically a distributed configuration, building a visual model of the elastic computing organization, and evaluating its effectiveness. Container-component model is integrated with the box-counting fractal analysis method and fractal control based on dynamic sampling of the workload. The example of fractal analysis and programming of the distributed gradient ascent algorithm is presented.

Chapter 22 analyses the problem of training qualified professionals in the field of information technology. It is natural to use software tools to support the educational process. Given the high entry threshold, the authors propose simple and accessible software tools that allow one to free up teacher time for effective student training [39]. The proposed solution does not pretend to the completeness, but it makes it possible to form control materials, conduct control measures of different levels, take into account the attendance of classes, the dynamics of the educational process, and maintain interaction with a group of students.

Chapter 23 introduces a system that collects massive amounts of texts from the Internet, analyzes them, builds the entity-event ontology, and presents it to the end-user as a knowledge base. It can be also viewed as an automatic text corpus processing method that allows using of classic statistical and data analysis methods by extracting domain-specific information from text. As extracted knowledge is highly structured and easily operated, it can be used by such methods without any further reference to the source texts.

Chapter 24 demonstrates the possibility of using deep learning methods to create a system for automatic 3D scanned objects classification. The choice of the appropriate deep architecture is based on a comparative analysis of existing SOTA models executed on different data sets. To select an object of interest from a large-scale space scan, the authors consider preprocessing methods: noise data filtering, reference plane deletion, and removing extraneous objects. An algorithm for the descriptive representation of three-dimensional models based on the modification of existing methods of ray casting is obtained. The possibility of using this descriptive representation to solve the problem of searching among 3D models and using search results for 3D scene auto-completion is demonstrated.

1.3 Conclusions

The book presents the research work of major experts in the field of numerical methods and computational mechanics, as well as, dynamic systems and information technology. Using methods of applied mathematics and numerical simulation, such diverse gas- and hydrodynamics processes have been studied as meteoroid flight in Earth's atmosphere, flow structure near descent module in Mars atmosphere, wave motions of fluids, the evolution of cluster beams, pulsating detonation waves, and flow in plasma thrusters. In the dynamics of solids, numerical methods have been developed to study the processes of fatigue damage development, stress state of teeth prosthesis, elastic waves' propagation during exploratory drilling on artificial ice island. Effective numerical algorithms have been proposed for solving dynamic systems with interval parameters, optimal control, and filtration problems. To solve most of the problems, researchers reported the use of parallel algorithms for multiprocessor high-performance computing systems. The book will be useful to scientists, researchers, undergraduate and postgraduate students specializing in the field of computational methods, parallel algorithms, gas dynamics, aerodynamics, hydrodynamics, plasma mechanics, multiphase flows, solid dynamics, dynamic systems, and information technologies.

References

1. Syzranova, N.G., Andrushchenko, V.A.: Simulation of the motion and destruction of bolides in the Earth's atmosphere. High Temp. **54**(3), 308–315 (2016)
2. Babakov, A.V.: Program package FLUX for the simulation of fundamental and applied problems of fluid dynamics. Comput. Math. Math. Phys. **56**(6), 1151–1161 (2016)
3. Babakov, A.V., Novikov, P.A.: Numerical simulation of unsteady vortex structures in near wake of poorly streamlined bodies on multiprocessor computer system. Comput. Math. Math. Phys. **51**(2), 245–250 (2011)
4. Gushchin, V., Kondakov, V.: On the Cabaret scheme for incompressible fluid flow problems with a free surface. Math. Models Comput. Simul. **11**(4), 499–508 (2019)
5. Gushchin, V., Smirnova, I.: The Splitting Scheme for Mathematical Modeling of the Mixed Region Dynamics in a Stratified Fluid. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Theory and Practice of Computational Mechanics. Smart Innovation, Systems and Technologies, vol. 173, pp. 11–21. Springer, Singapore (2020)
6. Maksimov, F.A., Churakov, D.A., Shevelev, Y.D.: Development of mathematical models and numerical methods for aerodynamic design on multiprocessor computers. Comput. Math. Math. Phys. **51**, 284–307 (2011)
7. Ivanov, I.E., Nazarov, V.S., Gidaspov, V.Yu., Kryukov, I.A.: Numerical simulation of the process of phase transitions in gas-dynamic flows in nozzles and jets. In: Jain L.C., Fa-vorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds) Advances in Theory and Practice of Computational Mechanics: Proceedings of the 21st International Conferenceon Computational Mechanics and Modern Applied Software Systems, SIST, vol. 173, pp. 133–150, Springer, Singapore (2019)
8. Nazarov, V.S., Ivanov, I.E., Kryukov, I.A., Gidaspov, V.U.: Modeling the dynamics of a gas-droplet substance in nozzles, taking into account the phase transition. J. Phys. Conf. Ser. **1250**, 012026.1–012026.10 (2019)

9. Moiseeva, D.S., Motorin, A.A., Stupitsky, E.L.: Assessment of the ionization effect during the distribution of a toroidal plasma bunch in a diluted atmosphere. Geomag. Aeron. **59**, 448–457 (2019)
10. Smirnov, E.V., Stupitskii, E.L.: Numerical simulation of the effect of rarefied plasma flow on the solid surface. J. Surf. Invest. X-ray Synchrotron Neutron Tech. **4**(6), 965–975 (2010)
11. Lopato, A.I., Utkin, P.S.: Toward second-order algorithm for the pulsating detonation wave modeling in the shock-attached frame. Combust. Sci. Technol. **188**, 1844–1856 (2016)
12. Lopato, A.I., Eremenko, A.G., Utkin, P.S., Gavrilov, D.A.: Numerical Simulation of Detonation Initiation: The Quest of Grid Resolution. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Theory and Practice of Computational Mechanics. SIST, vol. 173, pp. 79–89. Springer, Singapore (2020)
13. Aksenov, A.G., Chechetkin, V.M., Tishkin, V.F.: Godunov type method and the Shafranov's task for multi-temperature plasma. Math. Models Comput. Simul. **11**, 360–373 (2019)
14. Aksenov, A.G., Chechetkin, V.M.: Supernova explosion mechanism with the neutrinos and the collapse of the rotation core. Astron. Rep. **62**, 834–839 (2018)
15. Balashov, V., Cherkasova, M., Kruglov, K., Kudriavtsev, A., Masherov, P., Mogulkin, A., Obukhov, V., Riaby, V., Svotina, V.: Radio frequency source of a weakly expanding wedge-shaped xenon ion beam for contactless removal of large-sized space debris objects. Review of Scientific Instruments **88**(8), 083304.1–083304.5 (2017)
16. Obukhov, V., Pokryshkin, A., Popov, G., Svotina, V.: Stability of a Moving Control of a Service SC and a Space Debris Object at Impact on it by an Ion Beam. In: Razoumny, Yu.N., Graziani, F., Guerman, A.D., Contant J.-M. (eds.) Advances in the Astronautical Sciences DyCoSS'2017, vol. 161, pp. 665–675. Moscow, Russia (2017)
17. Vorobev, A.L., Elnikov, R.V.: Analysis of the structure of families of locally optimum solutions to the problem of the interplanetary transfer of a spacecraft with a low—thrust engine. Cosm. Res. **56**(5), 365–372 (2018)
18. Antropov, N.N., Akhmetzhanov, R.V., Bogatiy, A.V., Grishin, R.A., Kozhevnikov, V.V., Plokhikh, A.P., Popov, G.A., Khartov, S.A.: Experimental research of radio-frequency ion thruster. Therm. Eng. **63**(13), 957–963 (2016)
19. Burago, N.G., Nikitin, I.S.: Multiaxial Fatigue Criteria and Durability of Titanium Compressor Disks in low- and Giga-Cycle Fatigue Modes. In: Mathematical Modeling and Optimization of Complex Structures, pp. 117–130. Springer, Heidelberg (2016)
20. Burago, N.G., Nikitin, I.S., Nikitin, A.D., Stratula, B.A.: Algorithms for calculation damage processes. Frattura ed Integrità Strutturale **49**, 212–224 (2019)
21. Muratov, M.V., Petrov, I.B.: Application of mathematical fracture models to simulation of exploration seismology problems by the grid-characteristic method. Comput. Res. Model. **11**(6), 1077–1082 (2019)
22. Petrov, I.B., Muratov, M.V.: Application of the grid-characteristic method to the solution of direct problems in the seismic exploration of fractured formations (review). Math. Models Comput. Simul. **11**, 924–939 (2019)
23. Aruyunov, S.D., Grachev, D.I., Nikitin, A.D.: Mathematical modelling on the fracture of a laminar prosthesis basis under natural chewing loads. IOP Conf. Ser. Mater. Sci. Eng. **747**, 012065.1–012065.6 (2020)
24. Arutyunov, S.D., Grachev, D.I., Bagdasaryan, G.G., Nikitin, A.D.: Critical stress analysis for the basis of a denture prosthesis. IOP Conf. Ser. Mater. Sci. Eng. (in print) (2020)
25. Markov, YuG, Perepelkin, V.V., Filippova, A.S.: Analysis of the perturbed Chandler wobble of the Earth pole. Dokl. Phys. **62**(6), 318–322 (2017)
26. Krylov, S.S., Perepelkin, V.V., Filippova, A.S.: Long-period Lunar Perturbations in Earth Pole Oscillatory Process: Theory and Observations. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Theory and Practice of Computational Mechanics. SIST, vol. 173, pp. 315–331. Springer, Singapore (2020)
27. Panteleev, A.V., Karane, M.M.S.: Multi-agent Optimization Algorithms for a Single Class of Optimal Deterministic Control Systems. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Computational Mechanics and Numerical Simulation. SIST, vol. 173, pp. 271–291. Springer, Singapore (2020)

28. Panteleev, A.V., Pis'mennaya, V.A.: Application of a memetic algorithm for the optimal control of bunches of trajectories of nonlinear deterministic systems with incomplete feedback. J. Comput. Syst. Sci. Int. **57**(1), 25–36 (2018)
29. Chugai, K.N., Kosachev, I.M., Rybakov, K.A.: Approximate Filtering Methods in Continuous-Time Stochastic Systems. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Computational Mechanics and Numerical Simulation. Smart Innovation, Systems and Technologies, vol. 173, pp. 351–371. Springer, Singapore (2020)
30. Averina, T.A., Rybakov, K.A.: Using maximum cross section method for filtering jump-diffusion random processes. Rus. J. Numer. Anal. Math. Model.ing **35**(2), 55–67 (2020)
31. Kuzmina, N.M., Ridley, A.N.: A method for estimating the impact of the fare rules conditions. Civ. Aviat. High Technol. **224**(2), 138–146 (2016)
32. Bochkov, A.V., Zhigirev, N.N., Ridley, A.N.: Method of recovery of priority vector for alternatives under uncertainty or incomplete expert assessment. Dependability **17**(3), 41–48 (2017)
33. Morozov, AYu., Reviznikov, D.L.: Adaptive interpolation algorithm based on a kd-tree for numerical integration of systems of ordinary differential equations with interval initial conditions. Differ. Eqn. **54**(7), 945–956 (2018)
34. Morozov, AYu., Reviznikov, D.L., Gidaspov, VYu.: Adaptive interpolation algorithm based on a kd-tree for the problems of chemical kinetics with interval parameters. Math. Models Comput. Simul. **11**(4), 622–633 (2019)
35. Rybakov, K.A.: Modeling and analysis of output processes of linear continuous stochastic systems based on orthogonal expansions of random functions. J. Comput. Sys. Sc. Int. **59**(3), 322–337 (2020)
36. Rybakov, K.A.: Spectral method of analysis and optimal estimation in linear stochastic systems. Int. J. Model. Simul. Sci. Comput. **11**(3), 2050022 (2020)
37. Semenov, A.S.: Essentials of fractal programming. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Theory and Practice of Computational Mechanics: Proceedings of the 21st International Conference on Computational Mechanics and Modern Applied Software Systems, SIST, vol. 173, pp. 373–386, Springer, Singapore (2020)
38. Semenov, A.S.: Prototype based programming with fractal algebra. AIP Conf. Proc. **2181**, 020009 (2019)
39. Source codes Google-scripts. https://github.com/LevChern/eduprocess. Last accessed 4 Aug 2020

Part I
Computational Fluid Dynamics

Chapter 2
Aspects of Meteoroids Flight in the Earth's Atmosphere

Nina G. Syzranova and Viktor A. Andrushchenko

Abstract We study the motion of meteoroids in the Earth's atmosphere. It is shown that space bodies do not always fall on the Earth or explode and shatter into small fragments in the atmosphere. Instead, for certain aerodynamic characteristics and small angles of entry into the atmosphere, they may re-enter the outer space after traveling several thousand kilometres through the atmosphere.

2.1 Introduction

When space bodies pass into the Earth's atmosphere and move in it, various scenarios are possible. Large bodies (size greater than 100 m) usually reach the Earth's surface without losing much of their speed, forming impact craters, and can lead to catastrophic consequences. Small bodies (size less than 1 cm) burn completely at very high altitudes. Bodies of intermediate sizes are destroyed and burned at heights of ~20–40 km, causing bright flashes or exploding, breaking into fragments, forming shock waves and clouds of combustion products. Of particular interest are cases when meteor bodies or their fragments after the initial stage of falling in the atmosphere then go on an upward trajectory, and, only partially destroyed, return back to outer space. That is why, even when seismic effects appear as if from a fall, but in fact from the impact of an air explosion of meteoroid fragments, search expeditions often do not detect any impact craters or fallen meteorite matter.

Thus, on August 10, 1972, a flight through the atmosphere of a bright bolide detected by satellites of the US Air Force was registered [1]. Experts noted an unusually long flight path of the bolide in the atmosphere (about 1500 km). Witnesses even heard the thunderous sounds that indicated a low path of movement of the object.

N. G. Syzranova (✉) · V. A. Andrushchenko
Institute for Computer Aided Design of the RAS, 19/18, Vtoraya Brestskaya ul., Moscow 123056, Russian Federation
e-mail: nina-syzranova@ya.ru

V. A. Andrushchenko
e-mail: andrusviktor@ya.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_2

Everything pointed to the fact that the object was supposed to descend and fall to the ground, but its fall was never registered. This happened because the body was flying at a slight angle to the Earth's surface and "bounced" from the layers of the atmosphere, returned back to outer space [1, 2]. Estimates made in [3] show that such an intrusion into the atmosphere occurs quite rarely, and even more rarely, about once a century, such phenomena are observed. It is possible that such a meteor body was Tunguska (1908), the dynamics of which in the atmosphere is still a big mystery.

Thus, one of the important aspects of meteoritics is the study of the trajectories of meteor bodies under various conditions of entry into the Earth's atmosphere, which is the purpose of this chapter. Section 2.2 presents the basic equations for modeling the movement of large meteor bodies in the Earth's atmosphere. Section 2.3 presents the results of numerical calculations of the trajectories of meteor bodies at different angles of their entry into the atmosphere. Conclusions are presented in Sect. 2.4.

2.2 Basic Equations

To identify the main effects that accompany the movement of a large body in the atmosphere, we will study the body trajectory along which it moves under the influence of gravity and aerodynamic forces. In this case, the body mass will be assumed to be constant, meaning the mass loss caused by ablation will be considered insignificant, which is possible for large and durable meteoroids. In this case, changes in the speed of the meteoroid V and the angle of inclination of the velocity vector to the horizon θ are described by Eqs. 2.1–2.4 of the physical theory of meteors [4].

$$M\frac{\mathrm{d}V}{\mathrm{d}t} = \mathrm{Mg}\sin\theta - C_{\mathrm{D}}S\frac{\rho V^2}{2} \tag{2.1}$$

$$\mathrm{MV}\frac{\mathrm{d}\theta}{\mathrm{d}t} = \mathrm{Mg}\cos\theta - \frac{\mathrm{MV}^2\cos\theta}{R_{\mathrm{E}}+z} - C_N S\frac{\rho V^2}{2} \tag{2.2}$$

$$\frac{\mathrm{d}z}{\mathrm{d}t} = -V\sin\theta \tag{2.3}$$

$$\frac{\mathrm{d}L}{\mathrm{d}t} = V\cos\theta \tag{2.4}$$

Here C_{D}, C_N are the coefficients of drag and lift, respectively, S is the area of the body midsection, M is the mass of meteoroid, R_E is the Earth's radius, z is the altitude of the meteoric body above the Earth's surface, L, t are the range and time of the flight, respectively. The change in air density with height z is determined by the formula:

$$\rho = \rho_0 \exp(-z/h),$$

where ρ_0 is the atmospheric density with $z = 0$, h is the characteristic scale of altitude. In the Earth's atmosphere, for heights $z < 120$ km, the average value of $h = 7$ km. To solve the system of Eqs. 2.1–2.4, initial conditions are set for $t = 0 : V = V_e, \theta = \theta_e, L = 0, z_e = 100$ km.

We transform Eqs. 2.1–2.2 as follows:

$$\frac{\mathrm{d}V}{\mathrm{d}t} = g \sin\theta - \frac{\rho V^2}{2\lambda}, \tag{2.5}$$

$$\frac{\mathrm{d}\theta}{\mathrm{d}t} = \left(\frac{g}{V} - \frac{V}{R_\mathrm{E} + z}\right) \cos\theta - K\frac{\rho V}{2\lambda}. \tag{2.6}$$

Equations 2.5–2.6 contain two aerodynamic coefficients: the ballistic coefficient $\lambda = M/C_\mathrm{D}S$ and the aerodynamic quality $K = C_N/C_\mathrm{D}$. Moreover, the coefficient K for meteor bodies cannot exactly be equal to zero due to the imperfection of their shape, and its value for bodies of irregular geometric shape at hypersonic speeds can be more than 0.1 [5]. When estimating the ballistic coefficient λ for large meteor bodies with a mass of about 10^6 t at a density of 3 g/cm^3, it was found that it can reach $\lambda = 10^5$ kg/m^2. As a result, at high altitudes, the terms in Eqs. 2.5–2.6 representing the aerodynamic forces will be small, meaning that the atmosphere in this case has little effect on the movement of the body. This is the peculiarity of the movement of large meteor bodies in the atmosphere: the ability to penetrate the atmosphere.

2.3 Results of the Calculations

Using the system of Eqs. 2.1–2.4, calculations were made for a stony meteor body with a density $\rho_b = 3$ g/sm^3 and mass $M = 1 \times 10^6$t, entering the Earth's atmosphere at a speed of 30 km/s (these parameters presumably correspond to the Tunguska meteoroid) at different initial angles of entry of the body into the atmosphere. It is assumed that the coefficient of drag is equal to $C_D = 1$, and the value of the ballistic coefficient is $\lambda \approx 1.7 \times 10^5$ kg/m^2. Figure 2.1 shows the changes in the angle of inclination of the trajectory θ depending on the flight time for different initial angles of entry of the body into the atmosphere θ_e without taking into account the fragmentation of the considered body and assuming zero lift ($K = 0$).

It can be seen that the initial angle of entry into the atmosphere has a strong influence on the trajectory and flight time of the body. When $\theta_e \leq 9°$ the angle of the trajectory changes sign over time, and the trajectory becomes ascending. For the angle $\theta_e = 9°$ "ascent" begins on the 40th s of the flight, $\theta_e = 7°$ and $\theta_e = 5°$ the trajectory becomes ascending on the 30th s and the 20th s, respectively.

Data in Fig. 2.2 show how the height of the asteroid's flight changes depending on the flight time for different angles of its entry into the atmosphere. From the results shown, it can be seen that when $\theta_e > 9°$ the meteorite will fall to the Earth, and when $\theta_e \leq 9°$, starting from a certain height, its trajectory becomes ascending.

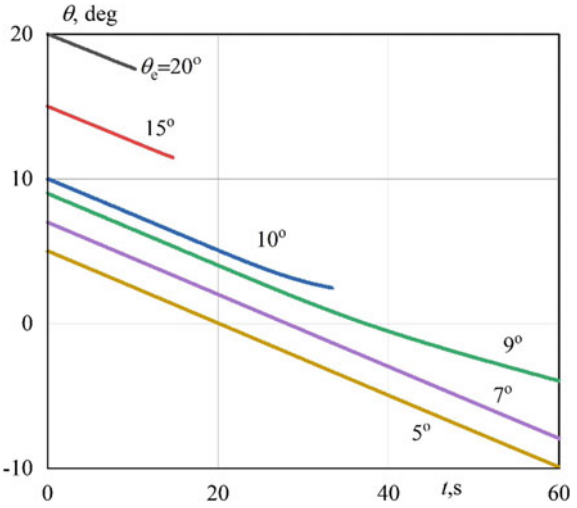


Fig. 2.1 The dependence of the trajectory angle θ on the flight time t of the meteoroid at different angles of entry θ_e into the atmosphere

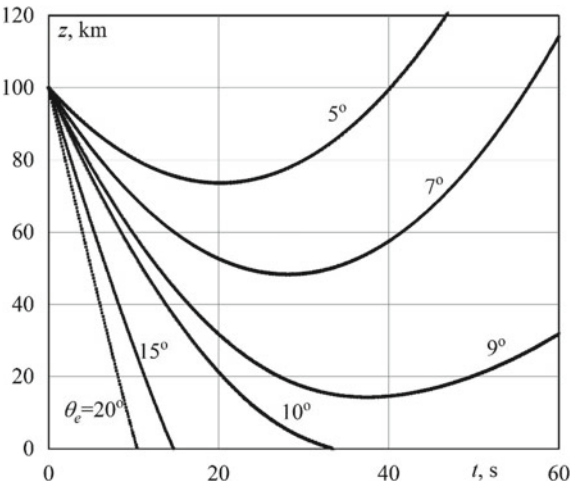


Fig. 2.2 The dependence of the flight altitude z on the flight time t at different angles of entry θ_e into the atmosphere

Figure 2.3 demonstrates how the body is decelerated down in the atmosphere at different angles of entry for a considered meteoroid. It can be seen that at the entry angles of 7° and 9°, the body does not reach the dense layers of the atmosphere, and either weakly slows down or does not lose speed at all.

It is interesting to study the movement of asteroids with the angles of entry into the atmosphere located in the interval $7° \leq \theta_e \leq 10°$. At the angle $\theta_e = 10°$ due to the shortness of the passage in the dense layers of the atmosphere, the meteoroid does not have time to decelerate down significantly and falls to the Earth's surface at the 34th s with a huge velocity of 16 km/s. This leads to an explosion and an almost instantaneous transition of kinetic energy of the meteoroid to mechanical and

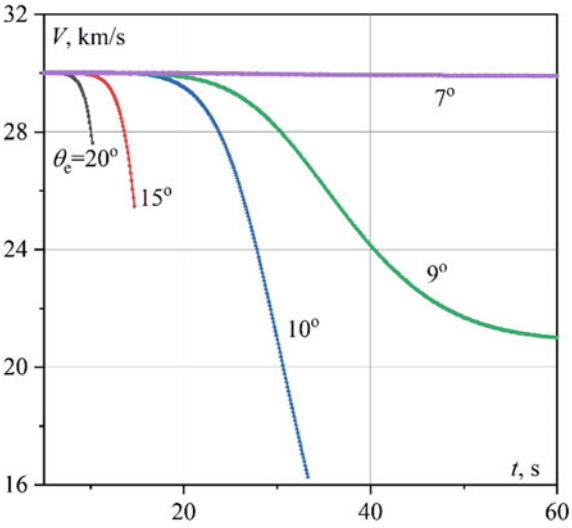


Fig. 2.3 The changing the speed of a meteor body depending on the flight time t for different entry angles θ_e

partially to thermal energy of the rocks surrounding the contact area of the surface, leading to their deformation and the formation of a crater of the funnel elongated in the direction of the fall [6] since this collision with the Earth occurs at an angle $\theta_e = 2.5°$ (see curve 10° in Fig. 2.1).

At the angle $\theta_e = 9°$, the celestial body penetrates only into the stratosphere and moves in it not even in a continuous, but in a transient flow regime, and during this time it loses a little less than a third of its initial velocity: $V = 21$ km/s when $t = 60$ s. At the angle $\theta_e = 7°$, moment of time $t = 60$ s, the body practically does not reach the dense layers of the atmosphere and almost does not lose velocity at all. At the values of velocities that are observed in Fig. 2.3, a rebound of a celestial body from the dense layers of the atmosphere can occur, like the rebound of a stone from the surface of water, and the meteoroid can fly into outer space.

Obviously, for a range of entry angles $7° \leq \theta_e \leq 10°$ can find a trajectory which is "soft landing", when the body during a long movement in the troposphere is strongly decelerated and at the same time the final part of its trajectory will be almost parallel to the surface. This fall leads to two possible consequences. If the body does not have time to slow down very much, then at a small angle θ_e, a long crater is formed in the direction of the fall, but shallow in depth [6]. As a result of such a meteorite fragment falling is the longest crater in Argentina, Rio Cuarto, which stretches 4.5 km in length, has a width of 1.1 km, and depth of only 7–8 m [7]. If the compacted mass of gas under the meteoroid is large enough, then the speed of its movement can fall significantly, and the impact speed will be close to zero, and such a meteorite will land as if on an air cushion. An example of such a landing is the famous Hoba meteorite (60t) in South Africa, which did not leave any noticeable traces when it fell on the surface [8].

The flight range of the meteoroid L for these entry angles is shown in Table 2.1 for flight paths, the range is calculated along the surface of the planet from the projection of the body's entry point into the atmosphere to the projection of its exit point at altitude $z = 100$ km, for the rest trajectories it is calculated from the projection of the body's entry point into the atmosphere to the point of falling off the body.

If the velocity of entry of a meteor body into the Earth's atmosphere is significantly lower than in the cases of data in Figs. 2.1, 2.2 and 2.3 ($V_e = 30$ km/s), then even a small entry angles it can reach the dense layers of the atmosphere, slow down to a speed less than the 2nd space velocity and eventually fall to the Earth. The parameters calculating such the movement of a body with mass $M = 1 \times 10^6$ t and angle of entry into the atmosphere $\theta_e = 7°$ at the initial velocity $V_e = 12$ km/s are shown in Figs. 2.4 and 2.5. From data in Fig. 2.4a, one can see how the angle of inclination of the trajectory decreases over time, and at some point its value becomes negative, but then again there is an increase in the angle of inclination of the trajectory to positive values and the trajectory of the meteoroid crosses the Earth's surface. The velocity of the body near the surface decreases to 4 km/s (Fig. 2.4b).

Table. 2.1 The flight range L of the meteoroid depending on the angle of entry θ_e

θ_e , deg	20	15	10	9	7	5
L, km	293	424	921	2207	1686	1205

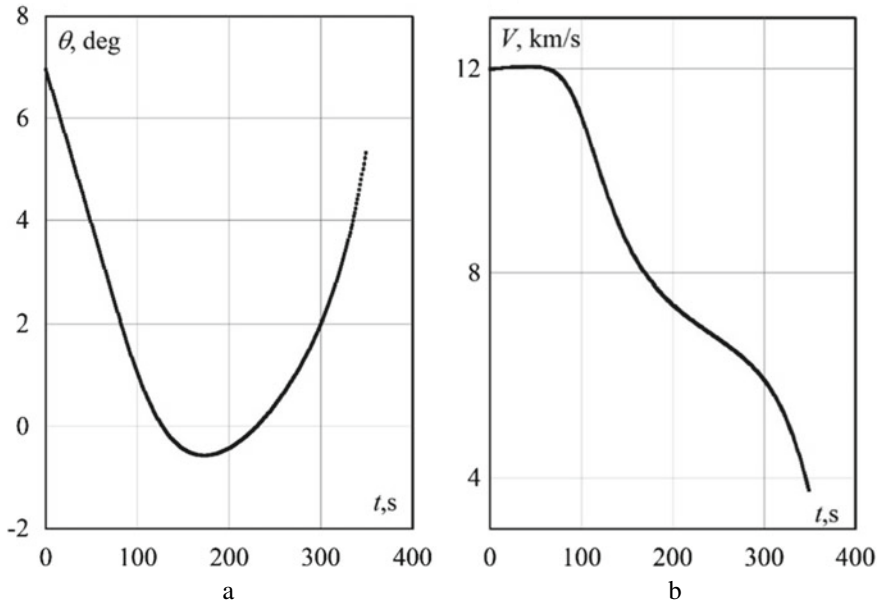


Fig. 2.4 The dependence of: **a** the trajectory angle, **b** velocity V of a meteor body on the flight time t for a body with mass $M = 1 \times 10^6$ t at $V_e = 12$ km/s and $\theta_e = 7°$

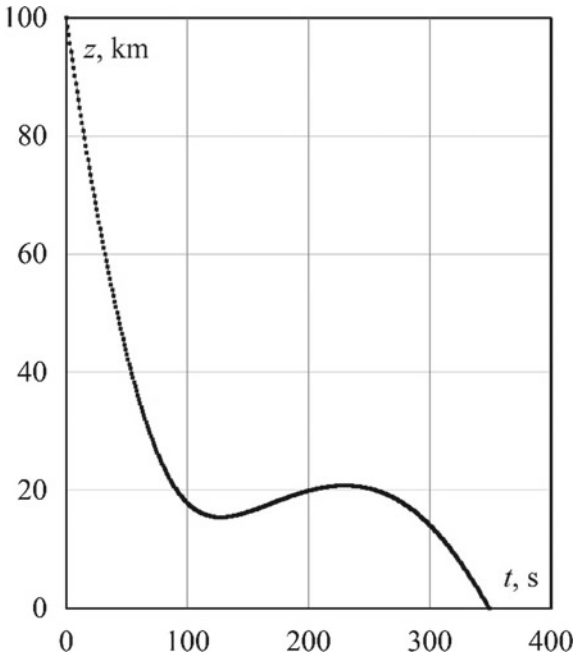


Fig. 2.5 The dependence of the flight altitude on the flight time t for the body mass $M = 1 \times 10^6$ t at $V_e = 12$ km/s and $\theta_e = 7°$

Data in Fig. 2.5 show how the flight height z of such a body changes depending on the flight time t. From these data, one can see the moments of time when the trajectory of the body becomes ascending, and when the stage of falling of the body occurs again. The value of the flight range in this case is $L \approx 3000$ km.

It should be noted that the data presented in Figs. 2.1, 2.2, 2.3, 2.4 and 2.5 are obtained at a zero value of the coefficient of aerodynamic quality: $K = 0$.

Figure 2.6 shows the calculation results of the dependence of the flight altitude z on the flight time t for the angle of entry of the body $\theta_e = 10°$ at different values of the coefficient K. It can be seen that at this value of the angle of entry into the atmosphere and $K = 0$, the trajectory crosses the Earth's surface; at $K \geq 0.1$ the body no longer crashes into the planet, but ricochets from the lower layers of the atmosphere. Moreover, the height of the ricocheting increases as coefficient K grows. In cases of negative values of the coefficient K, the trajectory curves in the other direction and the body falls to the Earth's surface in less time than in the case of $K = 0$.

In the case of negative values of the parameter K, the trajectory of the body, which $K = 0$ could be overflying, is curved in such a way that it falls to the surface of the Earth. This is shown by the curves in Fig. 2.7, which represents the calculation results for the input angle $\theta_e = 9°$ and values $K = 0, -0.1, -0.2$.

Thus, an imperfect geometric shape can have a significant impact on the trajectory of the meteoroid that is, the trajectory can "bend" up or down depending on the sign of the coefficient of aerodynamic quality.

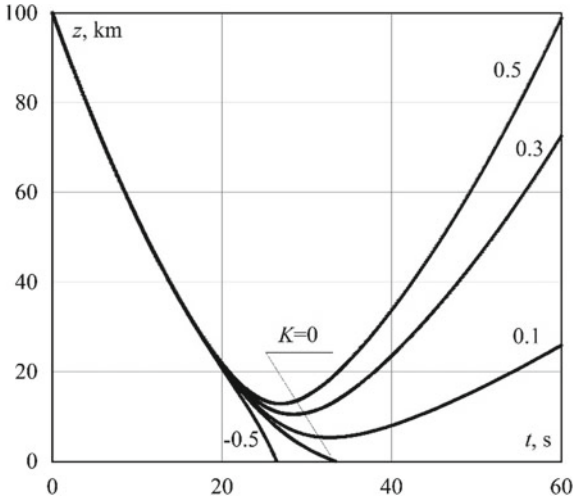


Fig. 2.6 The dependence of the flight altitude z on the flight time t at the angle of entry $\theta_e = 10°$ at different values of the aerodynamic quality coefficient K of the meteoroid

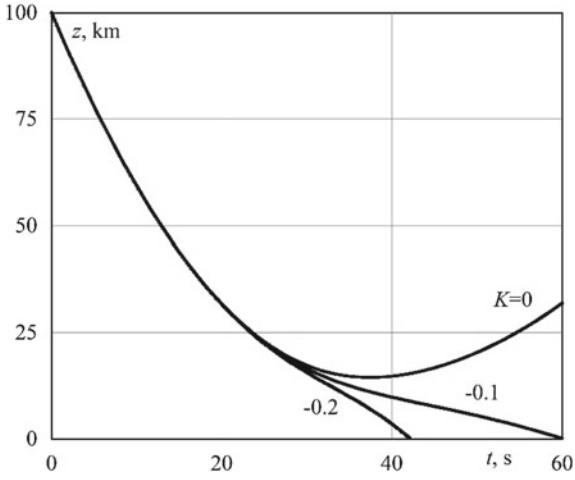


Fig. 2.7 The dependence of the flight altitude z on the flight time t at the angle of entry $\theta_e = 9°$ at different values of the aerodynamic quality coefficient K of the meteoroid

Let's consider the effect of its ballistic coefficient on the trajectory of a body. Suppose that a stone body enters the atmosphere with the velocity $V_e = 30$ km/s, but with a significantly lower mass as the previous one: $M \approx 60$ t, then the value of the ballistic coefficient will be $\lambda \approx 10^4$ kg/m^2.

Data in Fig. 2.8 show how the angle of the trajectory of the body changes depending on the flight time with two values of the ballistic coefficient at the angle of entry into the atmosphere $\theta_e = 9°$. It can be seen that for a body of lower mass ($\lambda \approx 10^4$ kg/m^2), the value of this angle is always positive and increases depending on the flight time (curve 1), whereas for a larger body with a higher value of the

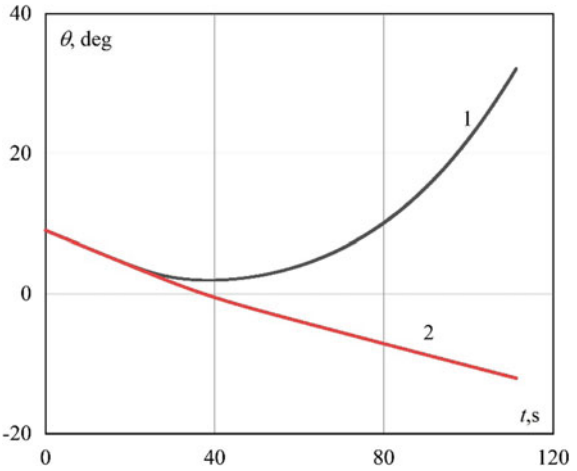


Fig. 2.8 The dependence of the trajectory angle θ on the flight time t of the meteoroid at different values of the ballistic coefficient λ: curve 1—$\lambda \approx 10^4$ kg/m^2, curve 2—$\lambda \approx 1.7 \times 10^5$ kg/m^2

ballistic coefficient ($\lambda \approx 1.7 \times 10^5$ kg/m^2), the value of angle changes sign, that is, trajectory of a body becomes ascending (curve 2).

As the results of the calculation show at $\lambda \approx 10^4$ kg/m^2, the trajectory becomes ascending, and the body acquires the ability to rebound from the atmosphere if the angle of entry into the atmosphere $\theta_e \leq 8°$, as shown by the curves in Fig. 2.9. These curves represent the dependence of the change in the height of the meteoroid flight on time for different angles of entry of the body into the atmosphere. It is seen that a decrease in the coefficient λ leads to a decrease in the critical angle of entry of the body into the atmosphere, below which flyby paths are possible.

The results obtained allow us to explain some of the effects of the Tunguska phenomenon in 1908. If the Tunguska meteoroid invaded the atmosphere at a small

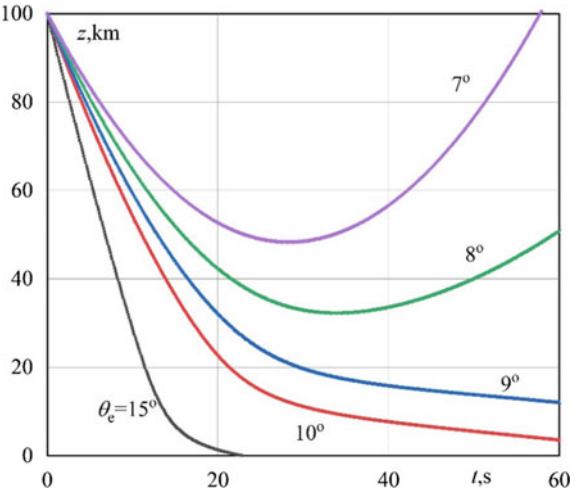


Fig. 2.9 The dependence of the flight altitude z on the flight time t for a body with a mass of 60t and the ballistic coefficient $\lambda \approx 10^4$ kg/m^2 for different angles of entry θ_e into the atmosphere

angle to the horizon ($\theta_e < 9°$), it could be a flyby. This did not exclude its fragmentation with explosions of some of its fragments in the atmosphere, leading to the collapse of the forest, but the main part of sufficiently large fragments could either fall far from the epicenter of the explosion or go into outer space. This assumption is also confirmed by the estimates given in [9]. The hypothesis we have considered allows us to explain the results of studying the proposed fall site of the Tunguska body by many expeditions: the absence of a crater and any material remnants of the meteoritic substance of this body. The results also show that the implementation of flight paths of meteoroids depends on a number of defining parameters of the phenomenon in the aggregate: speed, angle of entry of the body into the atmosphere, ballistic coefficient, and coefficient of aerodynamic quality. In addition, it is also important to take into account the fragmentation and destruction of the meteoroid under the influence of power and heat loads.

2.4 Conclusions

We simulate numerically the flight of large bodies in the Earth's atmosphere. Based on the model of a single body (no fragmentation), we determine the kinematic and physical characteristics necessary for a meteoroid to ascend in the atmosphere after its initial descend. We find that the key parameter for the possibility of such ascend is the angle of entry into the atmosphere. We compute the critical angles for a range of control parameters, i.e. the ballistic coefficient and the lift-to-drag ratio. Our results explain certain effects of the Tunguska event that took place in 1908.

References

1. Gordon, E., Bartky, C.D., Li, F. Wager, J.F.: Dynamics of a large meteor. In: 13th Aerospace Sciences Meeting, Pasadena, CA, U.S.A., AIAA Paper 75–14 (1975)
2. Ceplecha, Z.: Earth-grazing daylight fireball of August 10, 1972. Astron. Astrophys. **283**, 287–288 (1994)
3. Hartman, W.K.: The smaller bodies of the solar system. Sci. Am. **233**(3), 142–159 (1975)
4. Syzranova, N.G., Andrushchenko, V.A.: Simulation of the motion and destruction of bolides in the Earth's atmosphere. High Temp. **54**(3), 308–315 (2016)
5. Khokhryakov, V.A.: On the interaction of cosmic bodies with the atmospheres of planets. Space Res. **15**(2), 203–207 (1977) (in Russian)
6. Melosh, H.J.: Impact cratering: a geologic process. In: Oxford Monographs on Geology and Geophysics Series, 11 (1989)
7. Schultz, P.H., Zarate, M., Hames, W., Camilion, C. King, J.: A 3.3-Ma impact in Argentina and possible consequences. Science **282**, 2061–2063 (1998)

8. Spargo, P.E.: The history of the Hoba meteorite. Part III: Known and loved by all. Monthly Notes Astronom. Soc. Southern Africa **67**(11–12), 202–211 (2008)
9. Murzinov, I.V.: The problem of the century: where the tunguska meteorite flew. Cosmonaut. Rocket Sci. **83**(4), 65–72 (2015) (in Russian)

Chapter 3
Numerical Simulation of Flow Structure Near Descent Module in Mars Atmosphere

Alexander V. Babakov

Abstract Based on the conservative numerical method of flux, a hypersonic flow of a viscous heat-conducting gas is simulated near the landing module in Martian atmosphere. Special attention is paid to the study of the structure of non-stationary flow on the side and bottom surfaces of the module. Flow parameters data in the indicated areas are given. Vortex flow patterns near the descent module reflecting the spatial non-stationary nature of the flow are presented. Numerical modeling is implemented on multiprocessor supercomputers of cluster architecture.

3.1 Introduction

One method of studying aerodynamic characteristics of aerospace vehicles is numerical modeling. At the same time, the capabilities of this research method have significantly increased with the advent of multiprocessor supercomputers. It became possible to carry out modeling of spatial unsteady flows near objects of complex form based on more complete mathematical models.

The difficulty of numerical modeling of flows near objects of complex shape is associated with certain difficulties, primarily with the complex structure of the flow. The gas flow around poorly streamlined bodies, as a rule, is spatial, non-stationary, and has a vortex character. Moreover, for most aerodynamic problems of aerospace vehicles, the flow is characterized by high Reynolds numbers, which further complicates the numerical research. Vortex structures, non-stationary flow, and turbulence arising in such a flow have a significant impact on the aerodynamic characteristics of the vehicles.

From a computational point of view, the study of such flows requires the use of computational grids consisting of several tens of millions (or more) calculation

A. V. Babakov (✉)
Institute for Computer Aided Design of the RAS, 19/18, Vtoraya Brestskaya ul., Moscow 123056, Russian Federation
e-mail: avbabakov@mail.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_3

points. In addition, the study of such flows is complicated by the lack of an adequate turbulence model.

To determine the aerodynamic characteristics of the landing vehicles and calculate the descent trajectory, specialized software, on the basis of which a numerical study of the aerodynamics of segmental and conical shaped vehicles was carried out, is developed [1]. In [2], the aerodynamic characteristics of the descent module entering the Martian atmosphere were also calculated for the inviscid gas model, taking into account the physicochemical processes in the high-temperature shock layer.

In this work, a numerical simulation of the hypersonic flow around the descent module in the conditions of the Martian atmosphere is carried out. Modeling is based on a model of viscous, heat-conducting gas (the Navier-Stokes model). The main attention is paid to the study of non-stationary vortex motion occurring on the side surface and in near wake of the descent module.

The chapter is organized as follows. Section 3.2 provides a general description of the numerical method used. Section 3.3 is devoted to a statement of the problem for numerical simulation of the flow around the descent module. Section 3.4 presents the results of simulation of unsteady vortex structures of the flow near the lateral surface of the descent module. Flow patterns and data on gas-dynamic parameters are presented. The chapter ends with conclusions in Sect. 3.5.

3.2 Numerical Technique

Numerical studies are based on the non-stationary version of the conservative flux method [3, 4], which allows to calculate flow parameters in the entire integration area in a unified manner without allocation of features. The numerical model is written in the form of a finite-difference analogue of conservation laws written in integral form for each finite volume of the computational grid for each additive characteristic of the medium. The equations of motion in the methodology used are written in the Cartesian coordinate system for the Cartesian components of the vector quantities, regardless of the geometry of the problem and the type of computational grids used.

Numerical calculations were performed on the basis of the developed program complex “FLUX” [5, 6]. The method and program package are designed to study the spatially unsteady motion of a compressible gas at sub-, trans-, and supersonic speeds. The method allows to carry out the numerical studies of the aerodynamics of aerospace vehicles of complex shape in a wide range of determining parameters. The software package is based on parallel algorithms of the method and was implemented on modern supercomputer systems of cluster architecture.

The hardware, which is used for problem under consideration, includes 207 compute nodes with a peak performance of 521 TFlops consisting of 2 Xeon E5-2690 (Sandy Bridge) processors (64 Gb memory) and 2 Xeon Phi 7110X (KNC) processors (16 Gb memory) connected by networks based on InfiniBand FDR and Gigabit Ethernet.

3.3 Problem Statement

Numerical simulation of the flow around the descent module at hypersonic regime is considered. Numerical modeling is carried out on the basis of a model of viscous heat-conducting compressible gas (the Navier-Stokes model). The simulation is carried out in a three-dimensional non-stationary formulation using the gas-dynamic model of a perfect gas.

The descent module is a body of rotation, the frontal surface of which has the shape of a cone with a half-angle of 70° blunted over a sphere. The conical lateral surface has a spherical interface with the frontal surface.

The profile of the module and its general view are presented in Fig. 3.1. The shape of the module is close to the shape of the descent module presented in [7].

In Fig. 3.1 and in the following, all linear parameters are assigned to the radius R_0 of middle cross-section (further, mid-section).

Cartesian right-handed coordinate system *OXYZ* is used, which is associated with the descent module, the center of which coincides with the front point of the frontal surface. The axis *OX* is directed along the axis of symmetry of the module.

The integration area is bounded by the surface of the module and the outer boundary of the cylindrical shape of radius R_1 and length L_1. In the integration area, a non-uniform computational grid is introduced. The calculations were carried out on computational grids having exponential consolidation to the frontal, lateral, and bottom surfaces of the module. A fragment of the computational grid in a rarefied form is shown in Fig. 3.2.

Finite volumes are formed by splitting into constant steps at angular coordinate φ. The angle φ is counted in the *OYZ* plane from the positive direction of the *OY*

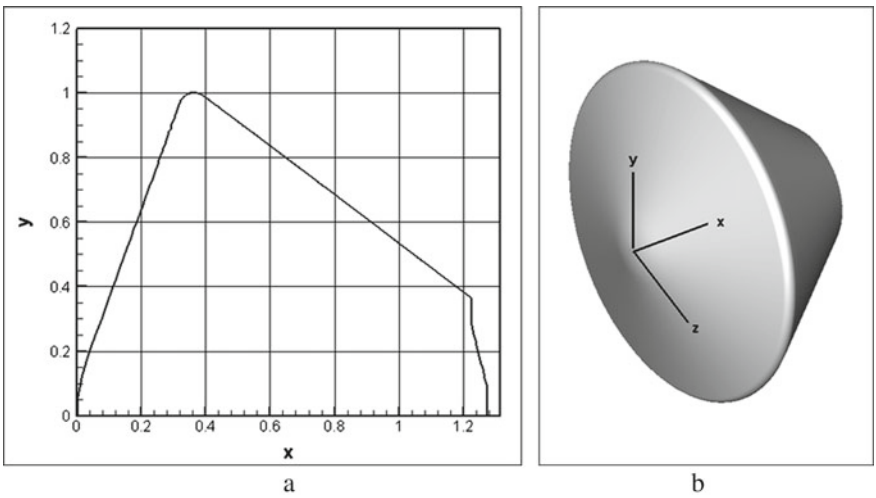


Fig. 3.1 Shape of the descent module: **a** profile, **b** general view

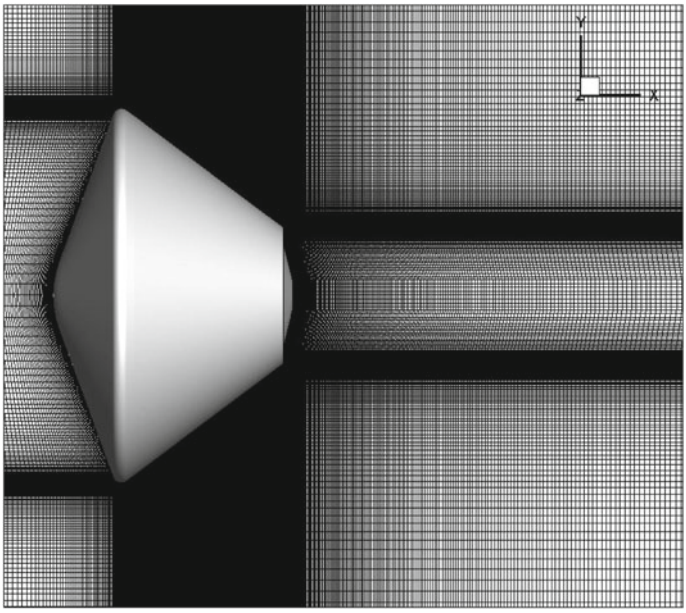


Fig. 3.2 Fragment of the computational grid in a rarefied form

axis toward the positive direction of the *OZ* axis. In the calculations, computational grids containing up to 20 million finite volumes were used.

For each finite volume, finite-difference analogues of the conservation laws of mass, momentum components, and total energy are written out. The system of finite-difference equations is closed by boundary conditions. On the left boundary $x = x_1$, the inflowing parameters are set (density $\rho = \rho_\infty$, temperature $T = T_\infty$, components of velocity $V_x = V_\infty$, $V_y = 0$, $V_z = 0$). On the right boundary $x = x_2$ or $r = R_1$, the "free" boundary conditions are set (parameters at the boundary are taken to be equal to those at the nearest calculated point). On the frontal, lateral, and bottom surfaces of the module, no-slip conditions ($V_x = V_y = V_z = 0$) and surface temperature T_w are specified. The pressure is determined by extrapolation from the stream.

In the calculations presented below, the parameters of the outer boundary of the integration domain took the following values: $R_1 = 2.0$, $x_1 = –0.5$, $x_2 = 10$, where R_1, x_1, x_2 are the dimensionless quantities.

3.4 Calculation Results

Hypersonic flow around the descent module at zero angle of attack is considered. The results presented below were obtained for the Mach number $M_\infty = 20$ and the Reynolds number $\mathrm{Re}_\infty = 1*10^6$ calculated from the inflow parameters and the radius

of the mid-section. Ratio of specific heat capacities of gas γ is taken equal to 1.335, corresponding to the atmosphere of the Mars. The surface temperature T_w of the apparatus was assumed to be 0.05 of the inflow adiabatic stagnation temperature.

In what follows, we will use the gas-dynamic variables in dimensionless form, namely the density ρ and temperature T will be related to the corresponding inflow parameters ρ_∞, T_∞, the velocity will be related to the inflow velocity V_∞, and the pressure p to $\rho_\infty V_\infty^2$ and time t to R_0/V_∞. The flow velocity V_∞ is directed along the positive direction of the OX axis.

As mentioned above, particular attention during the numerical calculations is paid to the study of flow properties near the lateral and bottom surfaces of the descent module. When integrated over time, the numerical solution acquires a non-stationary character. The flow near the lateral surface and in the near wake has a vortex non-stationary nature.

Thus, Fig. 3.3 shows the time behavior of the pressure coefficient $C_p = 2(p - p_\infty)$ at three points of the lateral surface in the meridional plane $\varphi = 0$: 1—mid-section, 2—in the middle of the lateral surface, and 3—at the point of interfacing of the lateral surface with the bottom section. The behavior of C_p indicates the steady state of the flow in the region of the mid-section (curve 1) and the unsteady nature of the flow on the lateral surface of the descent module (curves 2, 3).

This is confirmed by the flow patterns presented in Fig. 3.4, which shows the general view of the flow in the form of instantaneous streamlines and temperature field in the meridional plane $\varphi = 0$ and the bottom region.

The non-stationary and vortex nature of the flow near the lateral and bottom surfaces can be seen in Fig. 3.5, where instantaneous streamlines in the meridional plane $\varphi = 0$ are shown for various points of time against the background of the temperature field.

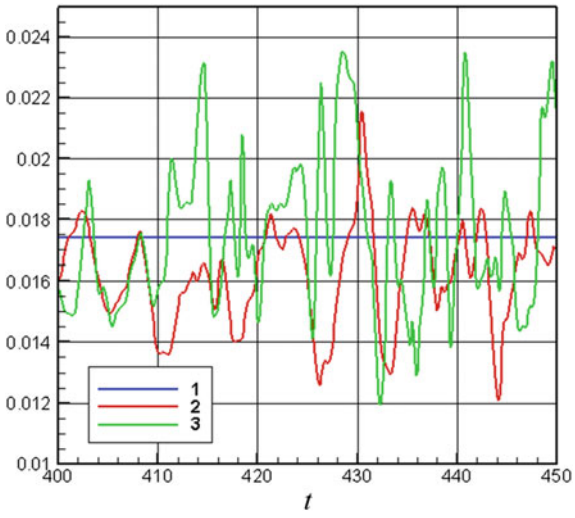


Fig. 3.3 Behavior in time of the pressure coefficient C_p on the lateral surface of the landing module: 1 $x = 0.34$, 2 $x = 0.8$, 3 $x = 1.22$

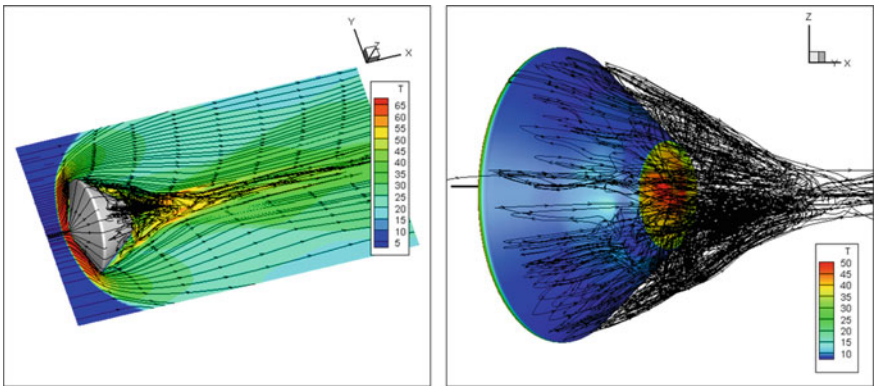


Fig. 3.4 Instantaneous streamlines and temperature field

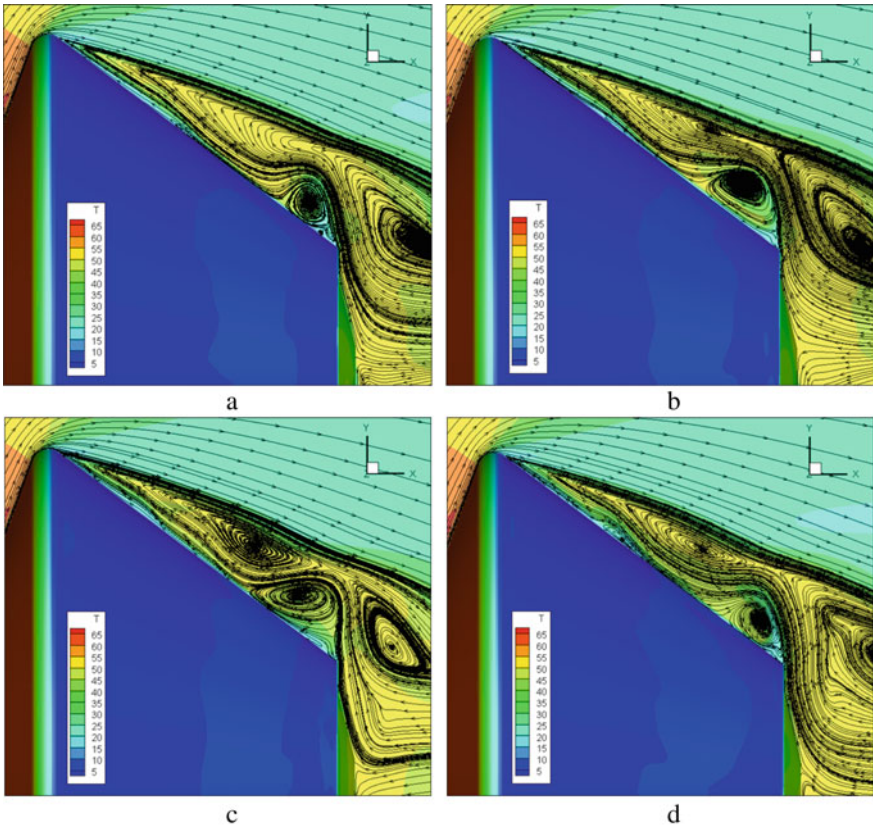


Fig. 3.5 Instantaneous streamlines in the meridional plane against the background of the temperature field at various points in time: **a** $t = 600$, **b** $t = 604$, **c** $t = 606$, **d** $t = 610$

A strong vortex movement occurs in the region of the lateral surface of the descent module. The unsteady nature of the flow in this region and near wake leads to non-stationary behavior of the aerodynamic characteristics. Thus, in particular, Fig. 3.6 shows the time behavior of the coefficient of moment m_z (which is calculated relative to the axis coinciding with the z axis). When calculating the coefficient of the moment acting on the descent module, the moment value was related to the $\rho_\infty V_\infty^2/2$, mid-section area and length L_1.

The non-stationary flow on the lateral surface of the descent module leads to a loss of axial symmetry of the flow. Thus, Fig. 3.7 shows the profiles of the pressure

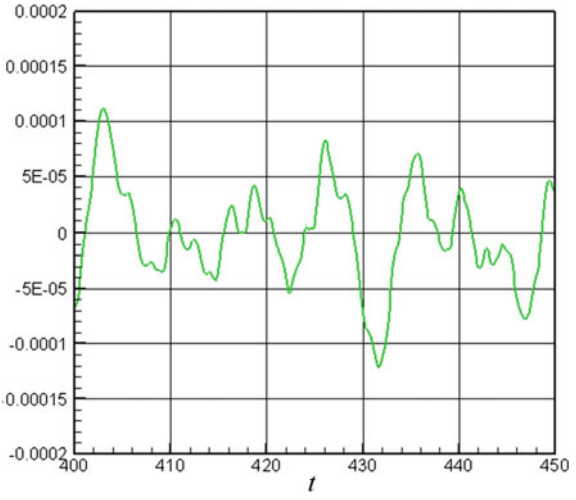


Fig. 3.6 Coefficient of moment m_z behavior over time

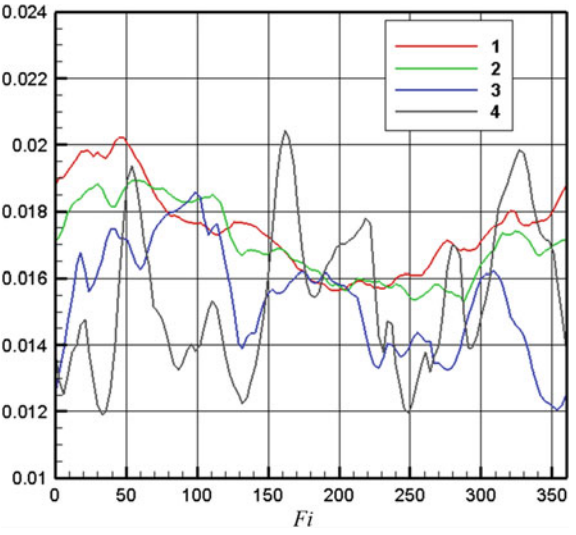


Fig. 3.7 Pressure coefficient C_p on the lateral surface of the descent module: 1 $x =$ 0.5, 2 $x = 0.6$, 3 $x = 0.8$, 4 $x = 1.0$

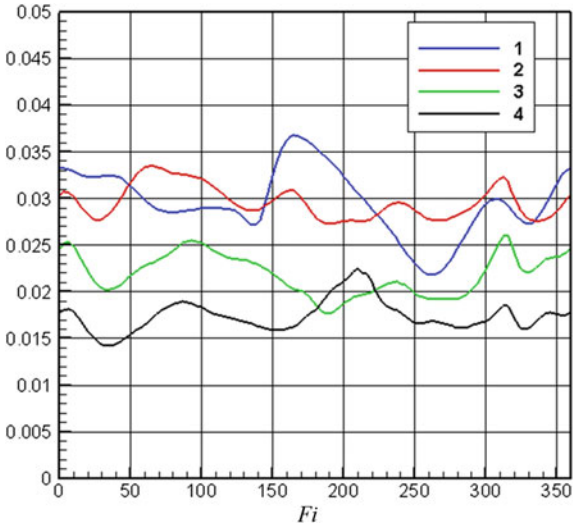


Fig. 3.8 Pressure coefficient C_p on the bottom surface of the descent module: 1 $r = 0.05$, 2 $r = 0.15$, 3 $r = 0.25$, 4 $r = 0.35$

coefficient C_p on the lateral surface of the module in various sections x for the time point $t = 600$.

For the bottom surface of the landing module for the time point $t = 600$, Fig. 3.8 shows plots of the pressure coefficient C_p versus the angular angle φ for various distances r from the axis of the vehicle.

In Fig. 3.9, for time $t = 600$, temperature fields are presented in different cross-sections along the axis of the descent module.

The above pictures give an idea of the temperature near the surface of the descent module and near wake. They also indicate the essentially non-stationary nature of the flow in this region. The area in which the unsteadiness of the flow is manifested is subsonic.

3.5 Conclusions

The results of the numerical simulation show that a developed unsteady vortex flow is realized on the lateral and bottom surfaces of the descent module, which affects the aerodynamic characteristics. It is important to note that the temperature near the lateral and bottom surfaces of module can reach large values. These features of the flow must be taken into account when designing new aerospace vehicles.

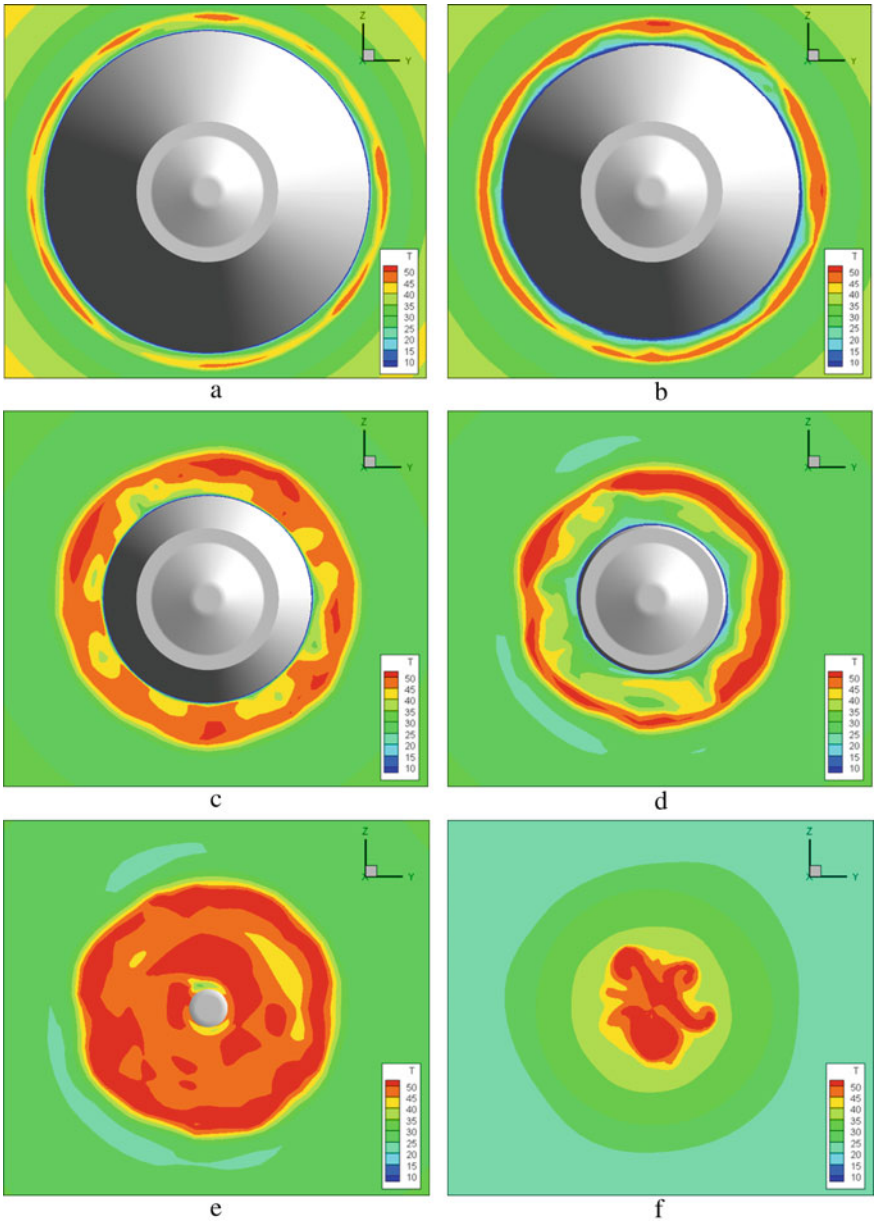


Fig. 3.9 Temperature fields in cross-sections along the axis of the descent module: **a** $x = 0.5$, **b** $x = 0.7$, **c** $x = 1.0$, **d** $x = 1.2$, **e** $x = 1.27$, **f** $x = 3.2$

Acknowledgements Calculations were carried out on the computational resources of the Joint Supercomputer Center of the Russian Academy of Sciences (JSCC RAS).

References

1. Golomazov, M.M., Finchenko, V.S., Ivankov, A.A., Shmatov, S.I.: Program package for the computer-aided aerodynamic design system for descent vehicles in planetary atmospheres. Sol. Syst. Res. **46**(7), 542–547 (2012)
2. Golomazov, M.M., Finchenko, V.S.: Aerodynamic design of a vehicle in the martian atmosphere under EXOMARS project. Sol. Syst. Res. **48**(7), 541–548 (2014)
3. Belotserkovskii, O.M., Severinov, L.I.: The conservative flow method and the calculation of the flow of a viscous heat-conducting gas past a body of finite size. Comput. Math. Math. Phys. **13**(2), 141–156 (1973)
4. Belotserkovskii, O.M., Babakov, A.V.: The simulation of the coherent vortex structures in the turbulent flows. Adv. Mech. **13**(3/4), 135–169 (1990)
5. Babakov, A.V.: Program package FLUX for the simulation of fundamental and applied problems of fluid dynamics. Comput. Math. Math. Phys. **56**(6), 1151–1161 (2016)
6. Babakov, A.V., Novikov, P.A.: Numerical simulation of unsteady vortex structures in near wake of poorly streamlined bodies on multiprocessor computer system. Comput. Math. Math. Phys. **51**(2), 245–250 (2011)
7. Finchenko, V.S., Ivankov, A.A., Shmatov, S.I., Mordvinkin, A.S.: Preliminary results of the calculated and experimental studies of the aerothermodynamic parameters of the EXOMARS landing module. Sol. Syst. Res. **49**(7), 557–568 (2015)

Chapter 4
Mathematical Modeling of Wave Motions of Fluids

Valentin A. Gushchin, Vasilii G. Kondakov, and Irina A. Smirnova

Abstract The study of wave movements of liquids is of interest both theoretically and practically. This can include flows with a free surface and flows with internal waves. For correct mathematical modeling of such flows, the finite-difference schemes of methods must have such properties as follows: high order of approximation, minimal scheme dissipation and dispersion, performance in a wide range of the Reynolds and the Froude numbers, and that is especially important the property of monotonicity. This chapter presents two approaches: splitting method for incompressible fluid flow (SMIF) method and compact accurately boundary adjusting high-resolution technique (CABARET) method, of course, whose finite-difference schemes have the properties listed above. A number of test tasks are considered and compared with theoretical, experimental data and calculations of other authors.

4.1 Introduction

In this work, two methods for problems of motion of fluids with a free surface and problems of internal waves' destruction in steadily stratified medium are considered. First method is known as SMIF [1, 2] or splitting on physical factor method for incompressible fluid flows. SMIF-based schemes have second order of approximation by both spatial and time steps. Schemes based on SMIF use mesh with spaced variables: when the velocity components normal to sides are set on the faces of cell,

V. A. Gushchin (✉) · I. A. Smirnova
Institute for Computer Aided Design of the RAS, 19/18, Vtoraya Brestskaya ul., Moscow 123056, Russian Federation
e-mail: gushchin@icad.org.ru

I. A. Smirnova
e-mail: o-ira@yandex.ru

V. G. Kondakov
Nuclear Safety Institute of the RAS, 52, Bol'shaya Tul'skaya ul., Moscow 115191, Russian Federation
e-mail: kondakov@ibrae.ac.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_4

and in the cell centers, the integral variables of density, pressure, salinity, temperature, etc. are defined. Second method is based on CABARET [3, 4] scheme or balance-characteristic approach. In contrast to SMIF, CABARET scheme uses a double set of variables both in faces and in the centers of cells. This allows CABARET scheme to use positive properties of characteristic approach such as solution of shock waves and rarefaction waves in the computational domain without introducing any restrictors and monotonizers, and the correct accounting of the flow at the boundary allows the schemes to remain conservative throughout the entire calculation time. These two methods are widely used in different domains of mathematical simulations: modeling of transport equations [4], equations of compressible gas [5], equations of incompressible flows in closed regions [6, 7], the Navier–Stokes equations of multicomponent gas dynamics [8], etc.

For the problems with a free surface, approximations of the Navier–Stokes equations in arbitrary Lagrange-Euler variables were obtained earlier for both SMIF scheme [9] and CABARET scheme [10]. Satisfactory results were also obtained in solving problems with stable stratification using the example of the problem of collapse of a spot [11–15] of a homogeneous fluid in the thick of a stratified medium.

In this chapter, we partially present the results of comparison of the application of two different numerical approaches for solving the problem of spot collapse: SMIF method and CABARET method in Sects. 4.2 and 4.3, respectively. Test problems are discussed in Sect. 4.4. Section 4.5 concludes the chapter.

4.2 SMIF Method

Numerical method for solving the problem of the dynamics of a spot (collapse) in a stably density-stratified fluid was discussed in [11]. This method can be used to investigate the flows of an inhomogeneous incompressible viscous fluid. The possibility of specifying the stratification of density and viscosity either by analytic formulas or by tables obtained by processing experimental data is foreseen that considerably widens the range of laminar flows considered. According to the model proposed in [16], the origin and development of turbulence in a stably density-stratified fluid are inseparable from internal waves and proceed as follows. Under the action of external forces, the internal waves of large size arise in the stratified fluid. As a result of their nonlinear interaction and subsequent breaking up or loss of stability, the regions of mixed fluid (spots) arise. These spots of mixed-up turbulent fluid evolve, gradually flattening (the collapse of turbulent spots), which in turn leads to the formation of new spots, and so on.

In the evolution of a spot, it is natural to consider three basic stages [16]:

- Initial stage: The motive force acting on the fluid particles situated inside a spot considerably exceeds the resistive forces. Intense internal waves are produced by the spot.

- Intermediate stationary stage: The motive force is mainly counterbalanced by the resistance of shape and the wave resistance due to the radiation of the internal waves. The increase of the horizontal size of the spot proceeds almost as a linear function of time, that is, the acceleration is negligibly small.
- Concluding viscous stage: The motive force is mainly counterbalanced by viscous drag. The horizontal size of the spot changes only slightly.

Later as a result of diffusion, the spot is mixed with the surrounding fluid and vanishes. Since that time, the simulation of such flows was undergone a lot of changes [11]. New physical and mathematical models have been proposed [17, 18], and the quality of methods designed for solving such problems has significantly improved [2, 9]. Furthermore, the progress in computing has been amazing. In this chapter, we want to adapt the mathematical model proposed in [17] and used for calculating the flow around a sphere and circular cylinder [17, 18] to the problem of spot collapse, which was earlier solved without taking into account the diffusion of the stratifying component [11, 19].

Consider the flat nonstationary problem about the flow occurring when a homogeneous fluid region A surrounded by a stably and continuously density-stratified fluid (for definiteness, the stratification is assumed to be linear) collapses in the vertical direction (Fig. 4.1). The flow develops in the homogeneous gravity field with the acceleration due to gravity g. The undisturbed linear density distribution [17]

$$\rho(x,\ y) = \rho_0(1 - \frac{y}{\Lambda} + s(x, y)) \tag{4.1}$$

is characterized by the stratification scale $\Lambda = \left|\frac{1}{\rho_0}\left(\frac{\partial\rho}{\partial y}\right)\right|^{-1}$, the buoyancy frequency $N = \sqrt{g/\Lambda}$, the buoyancy period $T_b = 2\pi/N$, $C = \Lambda/R_0$ is the ratio of scales, R_0

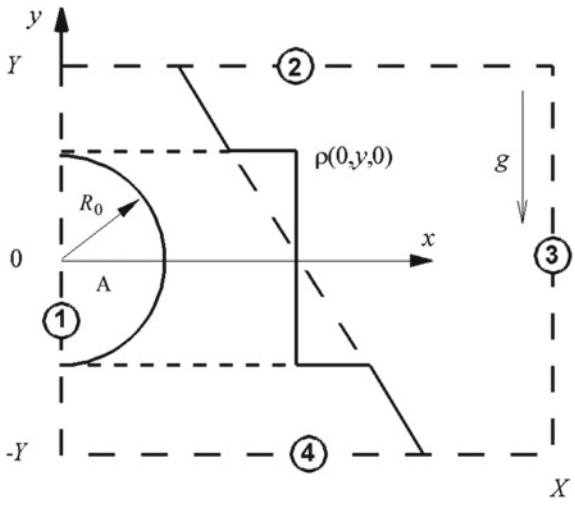


Fig. 4.1 Initial and boundary conditions for spot problem

is the spot radius, and s is the salinity perturbation (stratifying component), which includes the salt compression ratio.

We consider the plane unsteady problem of the flow which occurs when there is a collapse of the region A of homogeneous fluid, surrounded by a stably and continuously density-stratified fluid (see Fig. 4.1).

The Navier–Stokes equations in the Boussinesq approximation describing the flow of this type can be written as:

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} = -\nabla p + \frac{1}{\mathrm{Re}}\Delta \mathbf{v} + \frac{1}{\mathrm{Fr}} s \frac{\mathbf{g}}{g}, \tag{4.2}$$

$$\nabla \cdot \mathbf{v} = 0, \tag{4.3}$$

$$\frac{\partial s}{\partial t} + (\mathbf{v} \cdot \nabla)s = \frac{1}{\mathrm{Sc} \cdot \mathrm{Re}}\Delta s + \frac{v}{C}, \tag{4.4}$$

where $\mathbf{v}$ is the velocity vector with components u, v, respectively, along the x and y axes of a Cartesian coordinate system selected as indicated in Fig. 4.1, ρ is the density, p is the pressure minus hydrostatic one, s is the perturbation of salinity, the Reynolds number $\mathrm{Re} = \rho_0 {R_0}^2 N/\mu$, the Froude number $\mathrm{Fr} = R_0 N^2/g$, the Schmidt number $\mathrm{Sc} = \mu/\rho_0 k_\mathrm{s}$, k_s is the diffusion coefficient of salts, μ is dynamic viscosity coefficient, $\mathbf{g} = (0, -g)$, g is acceleration of free fall, ρ_0 is the density on the level $y = 0$, and $C = \Lambda/R_0$ is the scale ratio.

We assume that the initial time $t = 0$ the system on the plane $\mathbf{R}^2$ is at rest, i.e.,

$$u = 0, v = 0, (x, y) \in \mathbf{R}^2, \tag{4.5}$$

density of fluid at the spot A is

$$\rho = 1, (x, y) \in \mathrm{A}, \tag{4.6}$$

and outside of spot, i.e., in the area of $\mathbf{R}^2 \backslash \mathrm{A}$,

$$\rho = 1 - \frac{y}{C} + s, (x, y) \in \mathbf{R}^2 \backslash \mathrm{A}, \tag{4.7}$$

the perturbation of salinity is defined by Eq. 4.8.

$$s = \begin{cases} \frac{y}{C} \text{ if } (x, y) \in \mathrm{A} \\ 0 \text{ if } (x, y) \in \mathbf{R}^2 \backslash \mathrm{A} \end{cases} \tag{4.8}$$

As an initial approximation for pressure, necessary in solving the equation for pressure distribution is selected according to Eq. 4.9.

$$p = \begin{cases} -\frac{y}{\mathrm{Fr}} & \text{if } (x, y) \in \mathrm{A} \\ -\frac{y - y^2/2C}{\mathrm{Fr}} & \text{if } (x, y) \in \mathbf{R}^2 \backslash \mathrm{A} \end{cases} \tag{4.9}$$

As the pressure in the case of an incompressible fluid shall be determined with an accuracy of up to an arbitrary constant, without limiting the generality, we can select it to zero on level $y = 0$.

Effect of symmetry tasks concerning the plane $x = 0$ naturally seeks a solution in only one half-plane, for example, if $x \geq 0$. Solution will search in the rectangular area $\{x, y: 0 \leq x \leq X, -Y \leq y \leq Y\}$.

In the left boundary (line 1 in Fig. 4.1) this area are conditions of symmetry provided by Eq. 4.10.

$$u = 0 \quad \frac{\partial v}{\partial x} = \frac{\partial p}{\partial x} = \frac{\partial \rho}{\partial x} = \frac{\partial s}{\partial x} = 0 \tag{4.10}$$

The top (line 2), bottom (line 4), and right (line 3) borders should be chosen far enough away from the source of disturbance (from spots) so that setting any boundary conditions at these borders, which are necessary for the solution of the problem, not provided a significant influence on the flow.

To solve the task, we use one of the latest versions of a method of splitting by physical factors for research incompressible fluid flows (SMIF). Finite-difference scheme of this method possesses by properties such as a second-order approximation for the spatial variable, minimum scheme viscosity and dispersion, functioning in a wide range of Reynolds and Froude numbers, and more importantly when solving such problems as the monotony [2]. The splitting scheme and finite-difference scheme were described in detail in [14, 20].

4.3 CABARET Method

The study of waves in a fluid, as noted, is the subject of intensive theoretical and experimental research. In modes of practical interest, the nature of wave processes is determined by nonlinear vortex effects (e.g., wave overturning). All known analytical methods of solution are based on the assumption of potential flow. They make it possible to study the wave processes only until the waves start to overturn. After the waves start to overturn, this wave structure model becomes unacceptable. Physical experiments, on the other hand, are very complicated, laborious, and expensive. In addition, a number of rapidly occurring processes (particularly the overturning of waves) cannot be studied in detail in a physical experiment. In this respect, mathematical modeling of the corresponding physical processes becomes increasingly important. In these cases, numerical methods make it possible to obtain a more complete amount of information at lower costs and are often the only source of information about the flow field. The most general approach to the mathematical modeling of this

class of problems is to numerically integrate the complete nonstationary hydrodynamic equations. The known methods for calculating viscous incompressible fluid flows with a free surface do not allow obtaining highly accurate solutions near the free surface and in areas of large gradients of hydrodynamic flow parameters. In this respect, it is necessary to further develop methods of the numerical integration for the complete hydrodynamic equations with a free surface.

For solving such problems, in [10], it was proposed a new differential scheme based on CABARET method [3, 4]. The application of this approach for investigation of flows with a free surface may be seen in [21]. In [12], it was proposed a new finite-difference scheme based on CABARET technique for solving the spot problem. The motion of the medium is described by the Navier–Stokes equations and equation of continuity of the medium. The closure of the system occurs by adding the condition of zero divergence. The system of equations in dimensionless variables, where the characteristic linear size is equal to the spot radius, and the characteristic time is inversely proportional to the Brent-Väisälä frequency, is provided by Eq. 4.11.

$$\begin{cases} \frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial x} = \frac{1}{\mathrm{Re}}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \\ \frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial y} = \frac{1}{\mathrm{Re}}\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) - \frac{1}{\mathrm{Fr}} \\ \frac{\partial \rho}{\partial t} + u\frac{\partial \rho}{\partial x} + v\frac{\partial \rho}{\partial y} = 0 \\ \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \end{cases} \tag{4.11}$$

Here, the Reynolds and the Froude numbers are calculated by Eq. 4.12.

$$\mathrm{Re} = \frac{\rho_0 R_0^2 N}{\mu} \quad \mathrm{Fr} = \frac{R_0 N^2}{g} \tag{4.12}$$

The initial conditions for the problem with a spot are as follows (Fig. 4.1): In a circle with a radius of R_0, a density of ρ_0 is set or unitary in a dimensionless form, and a linear distribution $\rho = 1 - y \cdot \mathrm{Fr}$ is set around the circle such that the densities are the same at the center of the circle $y = 0$. At time $t = 0$, the velocities are zero, and the density is distributed as described above.

In a rectangular area $[-L_x; L_x]$ $[-L_y; L_y]$, an orthogonal grid set with nodes defined by Eq. 4.13 is formed.

$$\begin{cases} x_i = -L_x + 2L_x \frac{i}{N_x} \quad i = 0, N_x \\ y_j = -L_y + 2L_y \frac{j}{N_y} \quad j = 0, N_y \end{cases} \tag{4.13}$$

In the centers of the cells, we introduce conservative variables of density and velocity, and also in the centers of the faces of the cells, we introduce flux variables of density and velocity. Conservative cells are defined on half-integer layers in time, and flux variables are determined on integer layers in time. At the initial moment of time, conservative variables are initialized, and then the flux variables are calculated by any first-order difference scheme (e.g., a corner scheme) at the next level in time.

The system of Eq. 4.11 is reduced to a divergent form, using the continuity equations provided by Eq. 4.14.

$$\begin{aligned}
\frac{\partial \rho}{\partial t} &= -\left(\frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y}\right) \\
\frac{\partial \rho u}{\partial t} &= -\left(\frac{\partial \rho u^2}{\partial x} + \frac{\partial \rho uv}{\partial y}\right) + \frac{\partial p}{\partial x} + \frac{\rho}{\mathrm{Re}}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \\
\frac{\partial \rho v}{\partial t} &= -\left(\frac{\partial \rho uv}{\partial x} + \frac{\partial \rho v^2}{\partial y}\right) + \frac{\partial p}{\partial y} + \frac{\rho}{\mathrm{Re}}\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) - \frac{\rho}{\mathrm{Fr}}
\end{aligned} \tag{4.14}$$

The difference scheme can be written as follows:

$$\begin{aligned}
&\frac{\rho^{n+1/2} - \rho^{n-1/2}}{\tau} = -\nabla \cdot (\rho^n \mathbf{v}^n), \\
&\frac{\rho^{n+1/2}\tilde{u} - \rho^{n-1/2}u^{n-1/2}}{\tau} = -\nabla \cdot \left(\rho^n u^n \mathbf{v}^n\right) + \frac{\rho^{n-1/2}}{\mathrm{Re}} \Delta u^{n-1/2}, \\
&\frac{\rho^{n+1/2}\tilde{v} - \rho^{n-1/2}v^{n-1/2}}{\tau} = -\nabla \cdot \left(\rho^n v^n \mathbf{v}^n\right) + \frac{\rho^{n-1/2}}{\mathrm{Re}} \Delta v^{n-1/2} \\
&\qquad - \frac{\rho^{n-1/2} - 1 + y \cdot \mathrm{Fr}}{\mathrm{Fr}}, \\
&\nabla\left(\frac{1}{\rho^{n+1/2}} \nabla \delta p^{n+1/2}\right) = \frac{\nabla \cdot \tilde{\mathbf{v}}}{\tau}, \\
&\frac{\mathbf{v}^{n+1/2} - \tilde{\mathbf{v}}}{\tau} = -\frac{1}{\rho^{n+1/2}} \nabla \delta p^{n+1/2},
\end{aligned} \tag{4.15}$$

where the concept of overpressure is introduced by Eq. 4.16.

$$\begin{aligned}
&\delta p = p - p(y) \\
&p(y) = \int_y^{y_0} \rho(y) g \mathrm{d}y = p(y_0) - \rho_0 g\left(y - \frac{a}{2} y^2\right)
\end{aligned} \tag{4.16}$$

The values with integer indices in Eq. 4.15 refer to the flux variables, and the variables with half-integer indices belong to the conservative variables. As we see, the fifth difference expression of Eq. 4.15 is a modified Poisson equation, where the operator considers density inhomogeneity. This equation was solved using the parallel conjugate gradient method with a preconditioner in the form of the usual Laplace operator with constant density. The direct solver for the inversion of the Laplace operator was obtained by transforming the Fourier transforms of unknown pressure variables into twofold decomposition and equating the corresponding component of the right-hand decomposition.

Finally, the scheme ends by calculating the flow variables on the new time layer in the form of Eq. 4.17, where S is the index of the face of the flux variable, C is the index of the adjacent cell, from where the flow goes in the direction of the face of S, and S_{op} is the index of the face of the opposite face of S and belonging to cell C.

$$\boldsymbol{\psi}_S^{n+1} = 2\boldsymbol{\psi}_C^{n+1/2} - \boldsymbol{\psi}_{Sop}^{n} \quad \boldsymbol{\psi} = \begin{pmatrix} \rho \\ u \\ v \end{pmatrix} \tag{4.17}$$

Now let us consider the application of these two approaches for investigation of spot collapse in stratified fluid.

4.4 Test Problems

Hereinafter, the spot problem by SMIF method and CABARET method is presented in Sects. 4.4.1 and 4.4.2, respectively. Section 4.4.3 provides a comparison of results using these both methods.

4.4.1 Spot Problem by SMIF Method

Using SMIF method, the calculations were carried out in the field with dimensions $X = 10$, $Y = 5$, $R_0 = 1$ with the following coefficients and parameters: $\mu/\rho_0 = 0.01\ \mathrm{cm^2/s^{-1}}$, $k_s = 1.41 \times 10^{-5}\ \mathrm{cm^2/s^{-1}}$, $N = 1\ \mathrm{s^{-1}}$, $T_b = 2\pi$ s, $\Lambda = 10$ cm, $C = 10$, $\mathrm{Re} = 100$, $\mathrm{Fr} = 0.1$, $\mathrm{Sc} = 709.2$ that is close to the laboratory experimental conditions. As boundary conditions on the top, bottom, and right borders of the computational domain chosen resting state, i.e., $u = v = s = 0$. The computational domain was covered with a uniform grid with steps in both directions $\delta x = \delta y = 0.1$. With a view to verifying the correctness of program, the calculations in the absence of stain and on different grids were performed.

The time dependences of horizontal and vertical sizes of a spot are shown in Fig. 4.2.

4.4.2 Spot Problem by CABARET Method

In [12], authors used CABARET method for studying the problem of spot dynamics in a fluid that is stably stratified by density. In contrast SMIF method, a new difference scheme CABARET is solved by direct calculation elliptic equations for pressure by fast direct method [22].

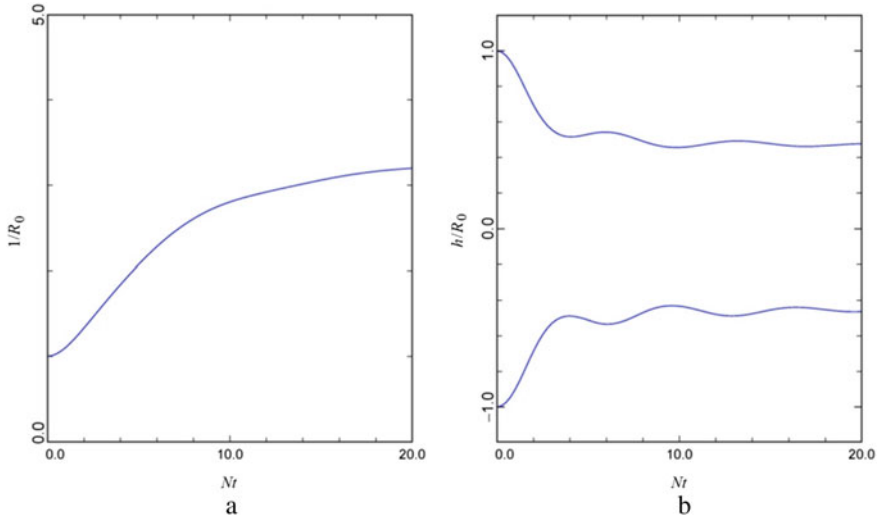


Fig. 4.2 Horizontal and vertical sizes of a spot in area 10×10. Re $= 100$, Fr $= 0.1$, Sc $= 709.2$ [15]: **a** horizontal spot size, **b** vertical spot size

Calculations of the problem with the collapse of the spot from [11] were carried out in the entire region with the numbers Re $= 100$ and Fr $= 0.1$. The dependences of the linear dimensions of the spot from time are shown in Fig. 4.3. The theoretical results [23, 24] are demonstrated for comparison.

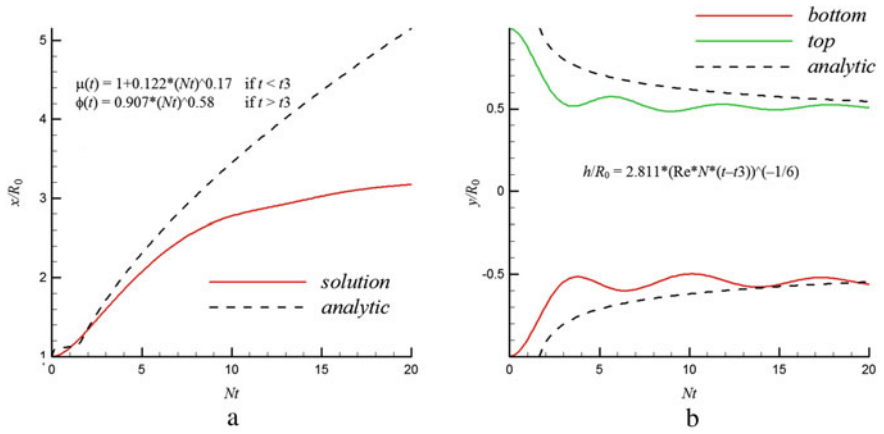


Fig. 4.3 Dependences of the linear dimensions of the spot from time: **a** growth of horizontal spot size, **b** vertical spot size setting

4.4.3 Comparison of Results Using SMIF and CABARET Methods

The comparison of results calculated by SMIF (solid lines) and CABARET (dotted lines) methods is demonstrated in Figs. 4.4 and 4.5, respectively. Calculations were carried out for Re = 100, Fr = 0.1. The sizes of spatial greed were the same in both cases. In Fig. 4.4, the dependence of horizontal size of spot from time is shown. The dependence of vertical sizes of spot from time is seen in Fig. 4.5. Some small

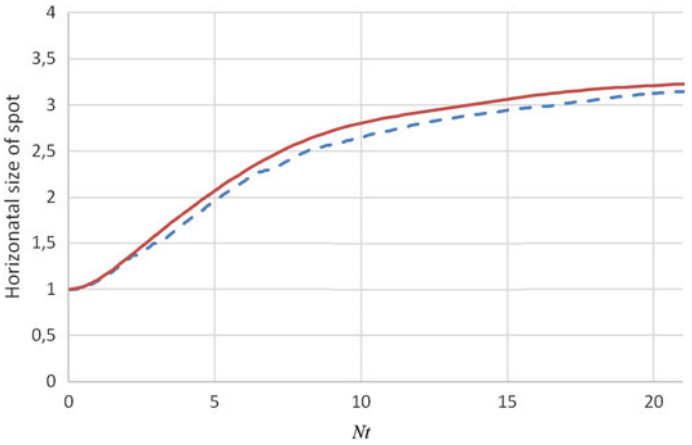


Fig. 4.4 Dependence of the horizontal size of the spot from time. CABARET area [−15; 15] × [−5;5], mesh 300 × 100; SMIF area [0;15] × [−5; 5], mesh 150 × 100

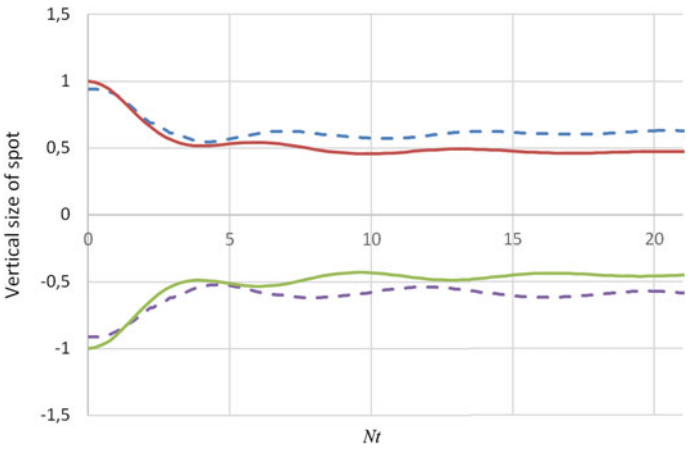


Fig. 4.5 Dependence of the vertical size of the spot from time. CABARET area [−15; 15] × [−5; 5], mesh 300 × 100; SMIF area [0; 15] × [−5; 5], mesh 150 × 100

distinction of results may be explained by using different physical models. In SMIF approach, the diffusion of stratification component (perturbation of salinity) is taken into account. In CABARET approach, the diffusion of stratification component—density is absent.

4.5 Conclusions

Two models of spot burst problem were considered: in the case of homogeneous density spot located in sharply stratified medium and in the case of salinity homogeneous spot in seawater stratified by salinity. The comparison of given results was carried out with fixed Froude numbers and Reynolds numbers. The case of homogeneous spot located inside viscous density-stratified medium was solved using CABARET method. The case with salinity diffusion inside salinity stratified medium was solved using SMIF method. Simulated domain in SMIF method is solved with plane $X = 0$ defined as symmetry boundary condition. Compared quantities (height and width of the spot time dependent) were dimensions of spot in horizontal and vertical direction time depending.

As we see from graph dependents of vertical spot dimension, top and bottom dimensions differ dynamically due to appearance of buoyancy force. Horizontal dimension of spot fairly correlates with theoretical estimation of spot dimension. From this, we can conclude that both methods are in sufficient agreement with other authors' works that deal with this task.

Acknowledgements This work was carried out in frame of the State Task of ICAD RAS.

References

1. Belotserkovskii, O., Gushchin, V., Shchennikov, V.: Use of the splitting method to solve problems of the dynamics of a viscous incompressible fluid. USSR Comput. Math. Math. Phys. **15**(1), 190–200 (1975)
2. Gushchin, V.: Family of quasi-monotonic finite-difference schemes of the second-order of approximation. Math. Models Comput. Simul. **8**, 487–496 (2016)
3. Goloviznin, V., Samarskii, A.: Some characteristics of finite difference scheme "CABARET". Math. Mod. **10**(1), 101–116 (1998) (in Russian)
4. Goloviznin, V., Samarskii, A.: Finite difference approximation of convective transport equation with space splitting time derivative. Math. Mod. **10**(1), 86–100 (1998) (in Russian)
5. Goloviznin, V., Karabasov, S., Kondakov, V.: Generalization of the CABARET scheme to two-dimensional orthogonal computational grids. Math. Models Comput. Simul. **6**(1), 56–79 (2014)
6. Glotov, V., Goloviznin, V.: CABARET scheme in velocity–pressure formulation for two-dimensional incompressible fluids. Comput. Math. Math. Phys. **53**(6), 721–735 (2013)
7. Gushchin, V., Matyushin, P.: Method SMIF for incompressible fluid flows modeling. In: Dimov, I., Faragó, I., Vulkov, L. (eds.) Numerical Analysis and Its Applications. NAA 2012. LNCS, vol. 8236, pp. 311–318. Springer, Berlin, Heidelberg (2013)

8. Glotov, V., Goloviznin, V., Kanaev, A., Kondakov, V., Kiselev, A.: CABARET scheme for modelling the stratification erosion in gas mixtures in hydrogen mitigation experiments for reactor safety. J. Phys.: Conf. Ser. **1392**(1), 012039.1–012039.7 (2019)
9. Belotserkovskii, O., Gushchin, V., Kon'shin, V.: The splitting method for investigating flows of a stratified liquid with a free surface. USSR Comput. Math. Math. Phys. **27**(2), 181–191 (1987)
10. Gushchin, V., Kondakov, V.: On the Cabaret scheme for incompressible fluid flow problems with a free surface. Math. Models Comput. Simul. **11**(4), 499–508 (2019)
11. Gushchin, V.: The splitting method for problems of the dynamics of an inhomogeneous viscous incompressible fluid. USSR Comput. Math. Math. Phys. **21**(4), 190–204 (1981)
12. Gushchin, V., Kondakov, V.: Solution of the problems of nonhomogeneous in-compressible fluid dynamics by the CABARET method. AIP Conf. Proc. **2164**(1), 120007.1–120007.8 (2019)
13. Gushchin, V., Kondakov, V.: Mathematical modeling of free-surface flows using multiprocessor computing systems. J. Phys.: Conf. Ser. **1392**(1), 012042.1–012042.6 (2019)
14. Gushchin, V., Smirnova, I.: On one approach for mathematical modeling of the mixed zones dynamics in a stratified fluid. AIP Conf. Proc. **2172**(1), 070017.1–070017.7 (2019)
15. Gushchin, V., Smirnova, I.: The splitting scheme for mathematical modeling of the mixed region dynamics in a stratified fluid. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Theory and Practice of Computational Mechanics. Smart Innovation, Systems and Technologies, vol. 173, pp. 11–21. Springer, Singapore (2020)
16. Barenblatt, G.: Dynamics of turbulent spots and intrusions in stably stratified fluid. Izv. Acad. Sci. USSR. Phys. Atmos. Ocean **14**(2), 195–206 (1978) (in Russian)
17. Gushchin, V., Mitkin, V., Rozhdestvenskaya, T., Chashechkin, Yu.D.: Numerical and experimental study of the fine structure of a stratified fluid flow over a circular cylinder. J. Appl. Mech. Tech. Phys. **48**, 34–43 (2007)
18. Gushchin, V., Matyushin, P.: Simulation and study of stratified flows around finite bodies. Comput. Math. Math. Phys. **56**, 1034–1047 (2016)
19. Gushchin, V., Kopysov, A.: The dynamics of a spherical mixing zone in a stratified fluid and its acoustic radiation. Comput. Math. Math. Phys. **31**(6), 51–60 (1991)
20. Gushchin, V., Smirnova, I.: Mathematical modeling of spot dynamics in a stratified medium. Comput. Math. Math. Phys. **60**(5), 879–894 (2020)
21. Gushchin, V., Kondakov, V.: One approach of solving tasks in the presence of free surface using a multiprocessor computing systems. In: Lirkov, I., Margenov, S. (eds.) Large-Scale Scientific Computing. LSSC 2019. LNCS, vol. 11958, pp. 324–331. Springer, Cham (2020)
22. Kuznetsov, Yu., Rossi, T.: Fast direct method for solving algebraic systems with separable symmetric band matrices. East-West J. Numer. Math. **4**(1), 53–68 (1996)
23. Wessel, W.: Numerical study of the collapse of a perturbation in an infinite density stratified fluid. Phys. Fluids **12**(12), 170–176 (1969)
24. Zatsepin, G., Fedorov, K.N., Voropaev, S.I., Pavlov, A.M.: Experimental research of collapse mixed spot in stratified fluid. Izv. Acad. Sci. USSR. Phys. Atmos. Ocean **14**(2), 234–237 (1978) (in Russian)

Chapter 5
Numerical Simulation of Taylor Vortex Flows Under the Periodicity Conditions

Fedor A. Maksimov

Abstract It is known from the experimental researches regarding the Taylor vortex flows between the rotating cylinders that a different number of pairs of the Taylor vortices can be formed within the one geometry. It means that different variants of the problem's solution are allowable. The simulation method with periodic boundary conditions on the edges of the cylinder's part was developed for the numerical research into the Taylor vortex flows**.** The results of the simulation of the flow for the various values of the periodicity sizes and different initial data are given.

5.1 Introduction

The theoretical research into the flows between the rotating cylinders assumes that they are endlessly long [1–3]. The experimental researches into the flows deal with the cylinders of the maximum length to reduce the edge effects. The simulation of a rather long cylinder requires the use of greater computational resources that inevitably lead to longer computational periods and relatively rough grid for the description of each vortex structure. The periodic structures with the scale of the distance order between the external cylinder and internal cylinder are formed along the axis of the cylinders in the Taylor vortex flow. To eliminate edge effects, it is possible to consider only a part of the cylinder, setting periodicity conditions on borders of the computational domain throughout the length. It requires the introduction of an additional dimension—the length L of the considered section of the cylinder. The dimension L that is artificially assigned in the research into the Taylor vortex flows determines their scale.

There are some experimental examples [4–6] when a different number of the Taylor vortices are formed in the same conditions, and, accordingly, the vortex pairs have a different size. In fact, the problem allows different stationary solutions while

F. A. Maksimov (✉)
Institute for Computer Aided Design of the RAS, 19/18, Vtoraya Brestskaya ul., Moscow 123056, Russian Federation
e-mail: f_a_maximov@mail.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_5

the implemented solution depends on different factors such as the mode of reaching the stationary solution, the geometric peculiarities that enable the formation of the vortices with the assigned size, and so on.

A large number of studies of the Taylor vortex flow have been performed using numerical simulations, for example, carried out recently [7–10]. The results of numerical modeling are in good agreement with theoretical and experimental data. This allows for the example of a flow with the Taylor vortices to study numerically the problem of bifurcation and non-uniqueness of the solution.

In this work, the calculations of the flow between rotating cylinders with various specified size of the periodicity L are performed. Section 5.2 describes a method for calculating three-dimensional flow based on a viscous gas model. In Sect. 5.3, it is shown that it is possible to construct a set of diverse solutions to the problem and estimates of the interval of admissible values of L. For some values L, at least two solutions can be constructed. In Sect. 5.4, the simulation method is used to analyze the flow of a viscous gas between rotating cylinders of different temperatures. Plane and axisymmetric solutions with the formation of vortex structures, as well as a fully three-dimensional flow from a combination of plane and axisymmetric flows, are obtained. Section 5.5 concludes the chapter.

5.2 The Simulation Method

The simulation uses the model of compressible viscid gas. The Navier–Stokes non-stationary equations for the three-dimensional flow of compressible gas in the dimensionless formed in the Cartesian coordinate system $\mathbf{X} = (x, y, z)$ are as follows:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial}{\partial x}(\mathbf{E} - \mathbf{E}_v) + \frac{\partial}{\partial y}(\mathbf{F} - \mathbf{F}_v) + \frac{\partial}{\partial z}(\mathbf{G} - \mathbf{G}_v) = 0,$$

where

$$\mathbf{U} = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ e \end{Bmatrix}, \quad \mathbf{E} = \begin{Bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ (e + p)u \end{Bmatrix}, \quad \mathbf{F} = \begin{Bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ (e + p)v \end{Bmatrix}, \quad \mathbf{G} = \begin{Bmatrix} \rho w \\ \rho uw \\ \rho wv \\ \rho w^2 + p \\ (e + p)w \end{Bmatrix}.$$

Here, t is the time, ρ is the density, (u, v, w) are the components of the velocity vector $\mathbf{V}$ in the (x, y, z) directions, respectively, p is the pressure, and e is the full energy of the gas volume's unit that can be determined for perfect gas as $e = \rho(\varepsilon + \frac{u^2+v^2+w^2}{2})$, where $\varepsilon = \frac{1}{\gamma-1}\frac{p}{\rho}$ is the internal gas energy, and γ is the heat capacity ratio with constant pressure and volume ($\gamma = 1.4$ in calculations). Then

$$\mathbf{E}_v = \begin{Bmatrix} 0 \\ \sigma_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ d_x \end{Bmatrix}, \quad \mathbf{F}_v = \begin{Bmatrix} 0 \\ \tau_{xy} \\ \sigma_{yy} \\ \tau_{yz} \\ d_y \end{Bmatrix}, \quad \mathbf{G}_v = \begin{Bmatrix} 0 \\ \tau_{xz} \\ \tau_{yz} \\ \sigma_{zz} \\ d_z \end{Bmatrix},$$

$$\begin{aligned} d_x &= u\sigma_{xx} + v\tau_{xy} + w\tau_{xz} + q_x, \\ d_y &= u\tau_{xy} + v\sigma_{yy} + w\sigma_{yz} + q_y, \\ d_z &= u\tau_{xz} + v\tau_{yz} + w\sigma_{zz} + q_z, \\ \operatorname{div} V &= \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}, \end{aligned}$$

$$\sigma_{xx} = \tfrac{\mu}{\mathrm{Re}}\left(2\tfrac{\partial u}{\partial x} - \tfrac{2}{3}\operatorname{div}\mathbf{V}\right), \quad \sigma_{yy} = \tfrac{\mu}{\mathrm{Re}}\left(2\tfrac{\partial v}{\partial y} - \tfrac{2}{3}\operatorname{div}\mathbf{V}\right), \quad \sigma_{zz} = \tfrac{\mu}{\mathrm{Re}}\left(2\tfrac{\partial w}{\partial z} - \tfrac{2}{3}\operatorname{div}\mathbf{V}\right),$$
$$\tau_{xy} = \tfrac{\mu}{\mathrm{Re}}\left(\tfrac{\partial u}{\partial x} + \tfrac{\partial v}{\partial y}\right), \quad \tau_{xz} = \tfrac{\mu}{\mathrm{Re}}\left(\tfrac{\partial u}{\partial x} + \tfrac{\partial w}{\partial z}\right), \quad \tau_{yz} = \tfrac{\mu}{\mathrm{Re}}\left(\tfrac{\partial v}{\partial y} + \tfrac{\partial w}{\partial z}\right),$$
$$q_x = \tfrac{\gamma}{\gamma-1}\tfrac{\mu}{\mathrm{Re\,Pr}}\tfrac{\partial T}{\partial x}, \quad q_y = \tfrac{\gamma}{\gamma-1}\tfrac{\mu}{\mathrm{Re\,Pr}}\tfrac{\partial T}{\partial y}, \quad q_z = \tfrac{\gamma}{\gamma-1}\tfrac{\mu}{\mathrm{Re\,Pr}}\tfrac{\partial T}{\partial z}.$$

The dimensionless variables are determined via the «′»-marked dimensional variables:

$$t = \sqrt{\tfrac{p'_o}{\rho'_o}}\tfrac{t'}{L'}, \quad \mathbf{X} = \tfrac{\mathbf{X}'}{L'}, \quad \mathbf{V} = \sqrt{\tfrac{\rho'_o}{p'_o}}\mathbf{V}',$$
$$\rho = \tfrac{\rho'}{\rho'_o}, \quad p = \tfrac{p'}{p'_o}, \quad T = \tfrac{T'}{T'_o}, \quad \mu = \tfrac{\mu'}{\mu'_o}.$$

The low index "$_o$" is the value of the parameter before the beginning of rotation. Here, L' is the characteristic dimension.

It is assumed that the Prandtl number $\mathrm{Pr} = \frac{\mu c_p}{\lambda}$ is constant. Here, μ, c_p, λ are the coefficients of heat capacity, viscidity, and heat conductivity ($\mathrm{Pr} = 0.72$ in calculations), respectively. $\mathrm{Re} = \frac{\sqrt{p'_0\rho'_0}L'}{\mu'_0}$ is the Reynolds number.

The system of differential equations is supplemented by the state equation: $p = \rho RT$, where T is the temperature, and R is the gas constant. The state equation in a dimensionless form is as follows: $p = \rho T$.

Thus, the complete system of the Navier–Stokes equations for perfect gas flows in the absence of external mass forces is given. It is assumed that the heat may enter the medium only as a result of thermal conductivity. The flows considered below are subsonic, and the compressibility effects are not of greater importance, but the use of the compressible gas model allows to apply the numerical method and the programs [11] developed for the simulation of viscous gas flows.

The use of the Cartesian coordinate system for the adequate description of the flows with complex topology in solving the problems by the finite-difference methods seems difficult for two reasons. One reason is that the interpolation procedures to get boundary conditions are necessary. Also there are some difficulties in the description of the computational grid because the computational grid is not rectangular. We will

move to the arbitrary curvilinear coordinate system. We will set the uniform grid for the difference approximation of the initial equations in the following coordinate system:

$$\tau = t, \quad \xi = \xi(x, y, z), \quad \eta = \eta(x, y, z), \quad \zeta = \zeta(x, y, z).$$

The use of the generalized transformation makes it possible to save the strictly conservative form of equations. The equations now look as follows:

$$\frac{\partial}{\partial \tau}\frac{\mathbf{U}}{J} + \frac{\partial}{\partial \xi}\frac{\overline{\mathbf{E}}}{J} + \frac{\partial}{\partial \eta}\frac{\overline{\mathbf{F}}}{J} + \frac{\partial}{\partial \varsigma}\frac{\overline{\mathbf{G}}}{J} = 0.$$

Here,

$$\begin{aligned}
\overline{\mathbf{E}} &= \xi_x(\mathbf{E} - \mathbf{E}_v) + \xi_y(\mathbf{F} - \mathbf{F}_v) + \xi_z(\mathbf{G} - \mathbf{G}_v),\\
\overline{\mathbf{F}} &= \eta_x(\mathbf{E} - \mathbf{E}_v) + \eta_y(\mathbf{F} - \mathbf{F}_v) + \eta_z(\mathbf{G} - \mathbf{G}_v),\\
\overline{\mathbf{G}} &= \varsigma_x(\mathbf{E} - \mathbf{E}_v) + \varsigma_y(\mathbf{F} - \mathbf{F}_v) + \varsigma_z(\mathbf{G} - \mathbf{G}_v).
\end{aligned}$$

The coefficients of the transformation matrix are determined by the following formulas:

$$\begin{aligned}
&\xi_x = J\left(\frac{\partial y}{\partial \eta}\frac{\partial z}{\partial \varsigma} - \frac{\partial y}{\partial \varsigma}\frac{\partial z}{\partial \eta}\right), \quad \xi_y = J\left(\frac{\partial z}{\partial \eta}\frac{\partial x}{\partial \varsigma} - \frac{\partial x}{\partial \eta}\frac{\partial z}{\partial \varsigma}\right), \quad \xi_z = J\left(\frac{\partial x}{\partial \eta}\frac{\partial y}{\partial \varsigma} - \frac{\partial y}{\partial \eta}\frac{\partial x}{\partial \varsigma}\right),\\
&\eta_x = J\left(\frac{\partial z}{\partial \xi}\frac{\partial y}{\partial \varsigma} - \frac{\partial y}{\partial \xi}\frac{\partial z}{\partial \varsigma}\right), \quad \eta_y = J\left(\frac{\partial x}{\partial \xi}\frac{\partial z}{\partial \varsigma} - \frac{\partial z}{\partial \xi}\frac{\partial x}{\partial \varsigma}\right), \quad \eta_z = J\left(\frac{\partial y}{\partial \xi}\frac{\partial x}{\partial \varsigma} - \frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \varsigma}\right),\\
&\varsigma_x = J\left(\frac{\partial y}{\partial \xi}\frac{\partial z}{\partial \eta} - \frac{\partial z}{\partial \xi}\frac{\partial y}{\partial \eta}\right), \quad \varsigma_y = J\left(\frac{\partial x}{\partial \eta}\frac{\partial z}{\partial \xi} - \frac{\partial x}{\partial \xi}\frac{\partial z}{\partial \eta}\right), \quad \varsigma_z = J\left(\frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial y}{\partial \xi}\frac{\partial x}{\partial \eta}\right).
\end{aligned}$$

Here, J is the Jacobian of the transformation determined by the formula:

$$J^{-1} = \frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta}\frac{\partial z}{\partial \varsigma} + \frac{\partial x}{\partial \varsigma}\frac{\partial y}{\partial \xi}\frac{\partial z}{\partial \eta} + \frac{\partial x}{\partial \eta}\frac{\partial y}{\partial \varsigma}\frac{\partial z}{\partial \xi} - \frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \varsigma}\frac{\partial z}{\partial \eta} - \frac{\partial x}{\partial \varsigma}\frac{\partial y}{\partial \eta}\frac{\partial z}{\partial \xi} - \frac{\partial x}{\partial \eta}\frac{\partial y}{\partial \xi}\frac{\partial z}{\partial \varsigma}.$$

The use of the generalized transformation makes it possible to construct the uniform grid in the form of a unit cube. The coefficients of the transformation matrix for the given distribution of nodes in the physical area are calculated with the use of difference formulas in accordance with the equations.

When composing the equations, it is assumed that the derivatives existing in the expressions for $\mathbf{E_v}$, $\mathbf{F_v}$, and $\mathbf{G_v}$ are transformed in accordance with the rules of the differentiation of the complex functions. These members are responsible for the presence of viscous forces. The method developed in [11] is used for the flow simulation. However, in contrast to the external aerodynamics case [11], it is necessary to take into account dissipative processes in all spatial directions when simulating the Taylor vortex flows.

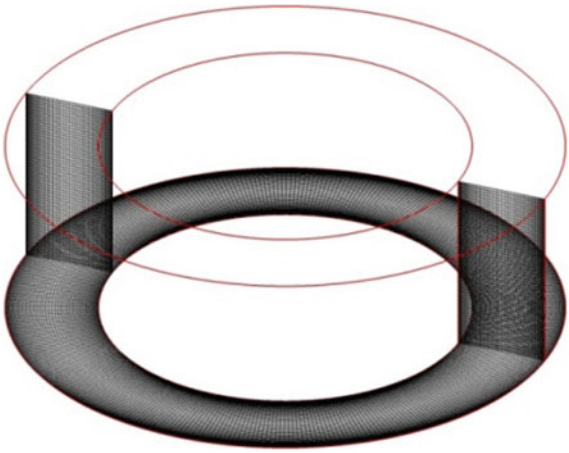


Fig. 5.1 Geometry of computational domain and grid in two sections

Figure 5.1 demonstrates the calculation domain between the cylinders and the grid in two sections.

The grid is uniform in the longitudinal and circumferential directions. Near the surfaces of the cylinders, the grid has a concentration along the normal to these surfaces, which makes it possible to improve the description of the velocity and temperature gradients in these areas. The calculations were made on the grid of 57 $\times$ 361 $\times$ 57 nodes (along the radius, in the circumferential direction and along the axis of the cylinders).

The adhesion conditions in accordance with the given cylinder rotation speed are set on the surface of the cylinders; the given temperature is fixed. The periodicity conditions are set along the axis of the cylinders on the edges of the computational domain. In this case, the gas-dynamic parameters for both edges at the two extreme layers are replaced by the values of the parameters in the inner layers from the other edge.

The calculations were made on the multiprocessor computing machinery, and parallelization was made with the help of the geometric decomposition of the computational domain.

5.3 Results of the Calculations

Let us examine the results of the numerical simulation using the example of calculating the flow between the endless cylinders: The radius of the inner cylinder is $r = 1$, and the radius of the outer cylinder is $R = 1.5$. The simulation calculates only a part of the endless cylinders: This is a section along the axis for the L length. Hereinafter, the L parameter is named the given periodicity size.

In the first variant of the calculations, the solution is found by the relaxation method from the initially given plane flow with a discontinuity in speed in the middle

between the cylinders. If the calculation for a certain L value is completed, then this solution can be used as initial data for some new value of L^* that is sufficiently close to the initial one. In this case, the values of the gas-dynamic functions in the nodes of the computational grid are assumed to be equal at the corresponding nodes for the previously calculated variant. This is the second calculation variant.

If the Reynolds number is rather small, then the Couette flow is the solution to the problem: This is a plane-parallel flow with a known velocity profile in the circumferential direction [12]. If to consider the conditions, in which the Couette flow is unstable [1], then the flat solution is to be destroyed due to instability in the first variant of the calculation, and then a stable regular solution with the Taylor vortices is to be arranged. In the second variant of the calculation, the flow structure with the Taylor vortices is actually set in the initial field, the solution is established in connection with the change in the periodicity size, and the stability of the flow structure to the change in parameter L is checked.

The inner cylinder rotates with the angular velocity ω, and the Reynolds number is $\mathrm{Re} = \frac{\omega \cdot r \cdot (R-r)}{\nu} = 200$, where ν is the kinematic viscosity coefficient. The linear velocity of the surface of the inner cylinder corresponded to the velocity with the Mach number $\approx$0.1, which does not lead to a significant change in the density in the flow and to the appearance of the compressibility effects. With the given Re number, the Taylor vortices are formed in the experiments [13]. The calculations for various preset values of the periodicity length L were made. As well as in the experiments, the Taylor vortices are formed in the numerical simulation with an adequate choice of parameter L. Figure 5.2 demonstrates an example of the solution with the Taylor vortices, where $L = 0.835$.

The flow is represented by the distribution of velocity along the axis of the cylinders in the plane passing through the axis of the cylinders. This velocity component is zero ($w = 0$), and the flow is plane in the Couette flow. When the Taylor vortices form, a "chess" structure with a periodic change of a sign along the axis of the cylinders and along the radius appears in the velocity distribution w. The flow is axisymmetric. Figure 5.2a demonstrates the distribution of the velocity w in the cross-section. It can be seen from Fig. 5.2b that the additionally superimposed isosurface of a constant value of this velocity is axisymmetric. Figure 5.2c demonstrates the spatial streamlines visualizing two Taylor vortices in a pair.

Figure 5.3 shows the number of pairs N of the Taylor vortices depending on the assigned periodicity size L. The set of the results obtained when using a flat field with the velocity discontinuity as the initial data corresponds to data 1 marked by large markers in the form of a circle. At $L < 0.35$, the Taylor vortices do not form, and the flow remains plane. At $0.4 < L < 1.2$, one pair of the vortices is formed; at $L > 1.2$, two pairs of the vortices are formed in the considered range up to $L = 2$.

The set of the results 2 corresponds to the case of motion with an increase of the size L, when using a solution with the Taylor vortices obtained at lower values of L as initial data. In this case, a solution with one pair of the Taylor vortices can be obtained for $L < 1.5$. This leads to significantly increase the range L, at which one pair of the Taylor vortices is formed.

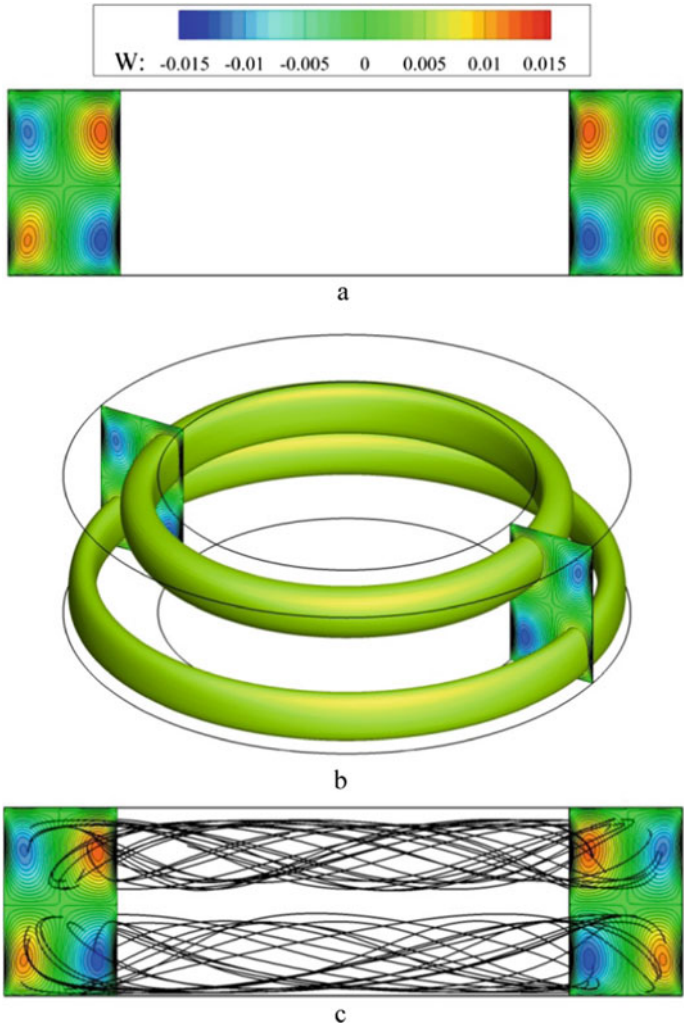


Fig. 5.2 Example of the solution with the Taylor vortices, $L = 0.835$: **a** distribution of the velocity in the cross-section, **b** superimposed isosurface, **c** spatial streamlines

Figure 5.4a, b shows an example of establishing a solution at $L = 0.835$, when using a solution with a discontinuity (a solid line) and with a close value of parameter L (a dashed line) as initial data. The establishment process is described by a change in the total kinetic energy of motion **E** in time in three directions: in the circumferential direction (Fig. 5.4a) and along the radius (black lines 1) and the axis of the cylinders (red lines 2) in Fig. 5.4b.

When calculating for the flow with a discontinuity, the plane character of the flow is preserved for some time, the kinetic energy along the axis of the cylinders and

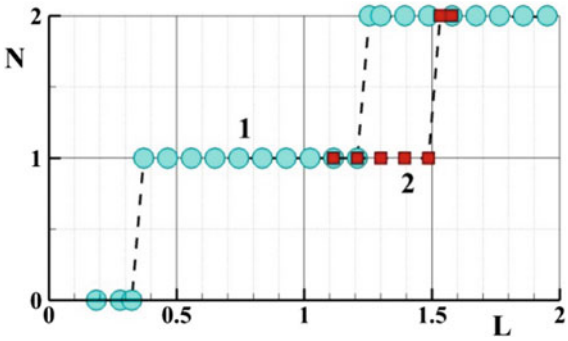


Fig. 5.3 Number N of the Taylor vortices depending on L

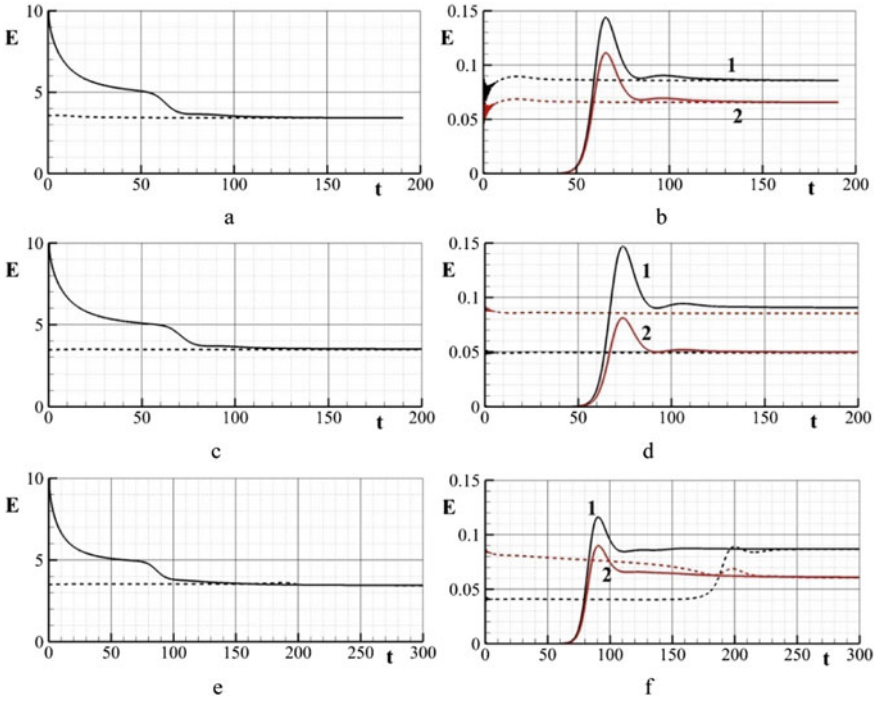


Fig. 5.4 Example of establishment of solution regarding total kinetic energy of motion in time: **a** circumferential direction, $L = 0.835$, **b** radius direction (black lines 1) and axis of the cylinders (red lines 2), $L = 1.393$, **c** circumferential direction, $L = 1.393$, **d** radius direction (black lines 1) and axis of the cylinders (red lines 2), $L = 0.835$, **e** circumferential direction, $L = 1.578$, **f** radius direction (black lines 1) and axis of the cylinders (red lines 2), $L = 1.578$

the radius is 0 at that time. Then instability develops, which leads to the formation of motion in all directions. The interesting fact is that the solution is reconstructed in a catastrophic manner, i.e., the given character of the flow is preserved over a certain time interval, then the surge of motion in the radial direction and along the axis of the cylinders occurs, and it is relatively small compared to the motion in the circumferential direction, but the solution changes qualitatively in the end. The flow in this case becomes three-dimensional. Then, upon reaching the stationary solution, the axisymmetric nature of the flow is established. When calculating with a change in the L parameter, the same solution is obtained; the process of establishing it is not accompanied by significant restructuring of the nature of the flow.

Figure 5.4c, d demonstrates an example of the restructuring of the flow at $L = 1.393$ in an analogous form. In this case, when calculating with an increase of parameter L, the nature of the flow with one pair of the vortices is determined by the initial field of the flow, and this flow character is preserved. When calculating for the flow with a discontinuity, two pairs of the Taylor vortices are formed. In fact, two different solutions are possible at one L value.

Figure 5.4e, f demonstrates an example of the restructuring of the flow at $L = 1.578$ in a similar form. In this case, when calculating with an increase of parameter L, the flow with one pair of the vortices re-forms into the flow with two pairs of the vortices. When calculating for the flow with a discontinuity, two pairs of the Taylor vortices are formed. The solutions are the same.

Figure 5.5 demonstrates the change in the Taylor vortex pattern depending on the size of the periodicity $L = 0.371$, $L = 0.557$, $L = 0.743$, $L = 0.929$, and $L = 1.114$ in cases when one pair of the vortices is formed regardless of the initial conditions. For visualization, the speed w along the axis of the cylinders is used. All the patterns were obtained in the same range of the variation w and in the same palette. A decrease in L to the value $L = 0.371$ leads to a decrease in the maximum velocity w. When the condition for the periodicity of the flow along the axis of the cylinders is preset, the

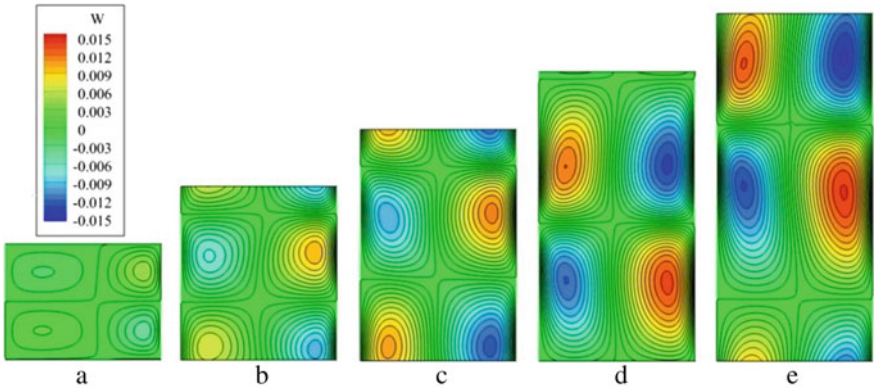


Fig. 5.5 Taylor vortices: **a** $L = 0.371$, **b** $L = 0.557$, **c** $L = 0.743$, **d** $L = 0.929$, **e** $L = 1.114$

Taylor vortices can shift relatively to the boundaries of the computational domain, which is observed in a number of calculations.

Figure 5.6 demonstrates the patterns of the Taylor vortices with $L = 1.300$, $L = 1.393$ when it is possible to get the flow with two pairs of the vortices (Fig. 5.6a) and one pair of the vortices (Fig. 5.6b). When using the field with the given flow structure as the initial condition, it is possible to delay the restructuring of the flow pattern.

The moment of resistance of the inner cylinder in the case of the Couette flow is determined by the expression $M = \frac{4\pi\mu\omega r^2 R^2}{R^2 - r^2}$ [12]. Let us introduce the dimensionless coefficient of the moment of friction resistance $C_m = \frac{M}{0.5\rho(\omega r)^2 \pi r^2 h}$. Here, h is the length of the cylinder. For the Couette solution, the coefficient C_m is determined by the expression $C_m = \frac{1}{\mathrm{Re}} \cdot \frac{8R^2}{(R+r)r}$. Figure 5.7 demonstrates the coefficient C_m depending on the given periodicity L in accordance with the calculation results. The 1 and 2 sets of results corresponded to the similar sets are shown in Fig. 5.7. If a sufficiently

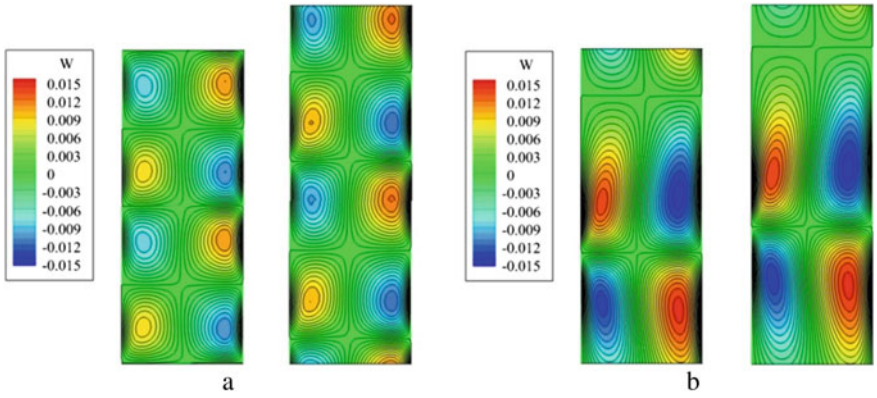


Fig. 5.6 Taylor vortices: **a** $L = 1.300$, **b** $L = 1.393$

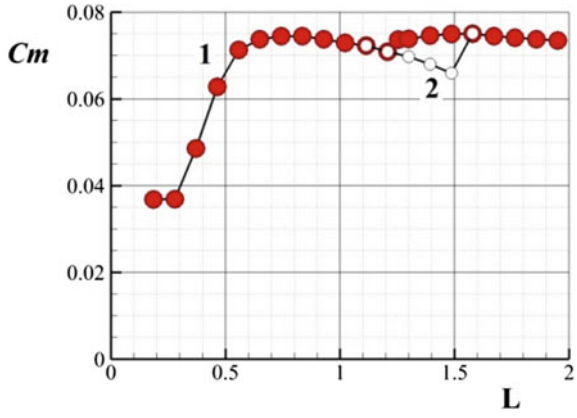


Fig. 5.7 Dependence of coefficient C_m from a periodicity size L

small periodicity size is set, the solution remains flat, and the Taylor vortices are not formed. The moment of friction in this case corresponds to the Couette solution $C_m = 0.036$. The requirement of periodicity at a small distance stabilizes the flow and in fact determines its two-dimensional character preventing the development of three-dimensional instabilities. In the conditions under consideration, the flat solution is conserved at $L < 0.35$.

At a sufficiently large value of L, regardless of the initial conditions, three-dimensional instabilities appear in the flow, and they subsequently formed into a regular flow with the Taylor vortices. One pair of the Taylor vortices is formed in the $0.4 \leq L \leq 1.2$ range between the cylinders. With the formation of the Taylor vortices, the friction moment increases significantly. There is the $L \approx 0.70$–0.85 value of the periodicity size at which the maximum value of the friction moment $C_m \approx 0.075$ is observed. Both a decrease and an increase in L lead to a decrease in the friction moment.

If to determine the flow structure by setting the initial conditions (with one pair of the vortices), then the structure can be preserved in a larger range of the periodicity size, and the friction moment decreases. Numerically, the solutions with the Taylor vortices of a relatively larger size are obtained. At $L \geq 1.2$, if the flow structure is not determined in some way, not one but two pairs of the vortices are formed in the flow area.

Figure 5.8 demonstrates the friction moment depending on the ratio of the periodicity size to the distance between the cylinders: $L/(R - r)$. In addition to the described results at $R = 1.5$, Fig. 5.8 shows the data obtained by the calculation for the values of the radius of the outer cylinder $R = 2.0$ and 1.25 ($r = 1$). When determining the Reynolds number, the distance $R - r$ between the cylinders is used as a characteristic size. For all the calculations, the Reynolds number is $\mathrm{Re} = 200$.

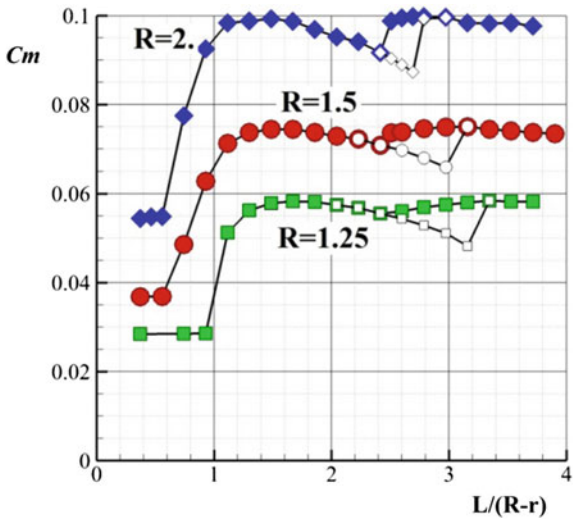


Fig. 5.8 Dependence of coefficient C_m from $L/(R - r)$ with $R = 2.0$, $R = 1.5$, and $R = 1.25$

Regardless of the geometry's parameter, when setting a sufficiently small size of the periodicity, the Couette flow is formed in the calculation. In this case, the calculated value of the friction moment is consistent with the theoretical value in accordance with the Couette solution ($C_m = 0.0533$, $C_m = 0.036$, and $C_m = 0.0278$ for $R = 2.0$, $R = 1.5$, and $R = 1.25$, respectively).

If the size of a pair of the vortices with the distance between the cylinders is correlated, then the size of a pair of the Taylor vortices can be from 1.0 to 2.2 of the distance between the cylinders. The estimate of the minimum size of a pair of the vortices decreases to 0.8 if to increase R. The estimate of the maximum size of a pair of the vortices is the same for three calculated geometry variants. Artificial pulling of the flow structure is possible. In this case, the size of a pair of the Taylor vortices can be up to 2.8–3.2 of the distance between the cylinders. Artificial pulling of the flow structure with the formation of the anomalously large Taylor vortices can be established, for example, by decreasing the distance between the cylinders in the already formed flow with the Taylor vortices.

The Taylor vortices, in fact, allow to actualize the maximum friction within the regular laminar flow. In accordance with this fact, the choice of the solution with the maximum friction, which is determined by the maximum of the friction, is the most correct one for the real flow. In accordance with the calculation results, this maximum exists, and in this case, the optimal size of a pair of the vortices is from 1.4 to 1.8 of the distance between the cylinders. As the distance between the cylinders decreases, the optimal size of the vortex pair increases.

In accordance with the experimental studies [13], the value of the coefficient of the friction moment at $\mathrm{Re} = 200$ is $C_m \approx 0.095$ for $R = 2$, and the calculation results satisfactorily conform to the experiment.

5.4 The Heat Exchange Between Rotating Cylinders

The flow of viscous gas between the rotating cylinders with different temperature is considered. The outer cylinder is heated, and therefore, the gas near its surface has a lower density. Under the influence of centrifugal force in the system of the rotating cylinders, the Rayleigh–Taylor instability will develop.

The radius R of the outer cylinder is two times larger than the radius r of the inner cylinder that rotates with the angular velocity ω. The Reynolds number is determined by the expression $\mathrm{Re} = \omega r \cdot (R - r)/\nu$. The Re critical number, at which the Couette flow reforms into the flow with the Taylor vortices, is approximately $\mathrm{Re}^* \approx 70$ [13] for a fixed and unheated external cylinder for the given geometry. For the formation of the Rayleigh–Taylor instability, the temperature of the outer cylinder was set two times higher than the temperature of the inner cylinder. The rotation of the outer cylinder varied from the rest state (the outer cylinder does not rotate) to the rotation with the same angular velocity as the inner cylinder, while the linear velocity of the surface of the outer cylinder was two times higher than the linear velocity of the inner cylinder. One of the intermediate rotation variants of the outer cylinder

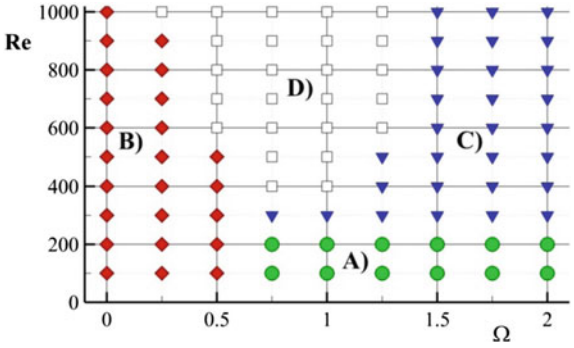


Fig. 5.9 Map of the flow modes, where A is the Couette-type flow, B is the axisymmetric flow with the Taylor vortices, C is the plane flow with the vortex structures parallel to the axis of the cylinders, D is three-dimensional flow with the Taylor vortices and the vortex structures parallel to the axis of the cylinders

corresponded to half the angular velocity of the rotation of the inner cylinder, while the linear surface velocities of the outer and inner cylinders were the same. The parameters that determine the calculation conditions are the Re number and the ratio Ω of the linear velocity of the surface of the outer cylinder to the linear velocity of the surface of the inner cylinder. The Re number was considered in the range from 100 to 1000. The ratio Ω of the linear velocities of the surfaces of the cylinders was from 0 to 2. The solution is obtained numerically by the relaxation method from a state of rest with a discontinuity in density. The size of the periodicity was assumed to be 1.85.

According to the calculation results, various types of the flow are formed depending on the parameters. In Fig. 5.9, the map of the flow modes in the parametric area (Ω, Re) is shown with four obtained types of flows.

The Couette-type flow is formed at small values of the Re number and at sufficiently high ratio Ω. In this case, the flow parameters change only depending on the distance to the axis of the cylinders. Figure 5.10a demonstrates the density distribution at $\mathrm{Re} = 200$, $\Omega = 0.75$. This is an example of the Couette-type flow.

For small values of the parameter Ω and for the sufficiently large values of the Re number, the Taylor vortices form in the flow. Figure 5.10b shows a typical flow pattern for the formation of the flow with the Taylor vortices in the form of the density distribution at $\mathrm{Re} = 200$, $\Omega = 0.50$. A decrease in the rotation speed of the outer cylinder due to an increase in the difference of centrifugal forces between the layers near the inner and outer cylinders leads to the development of the three-dimensional instability and the formation of the Taylor vortices.

The Taylor vortices are the effect of the three-dimensional instability of the Couette flow at the sufficiently large Reynolds number with a weakly rotating external cylinder. With a sufficiently high value of the parameter Ω, and in fact with a decrease in the difference of the centrifugal forces between the layers near the rotating inner

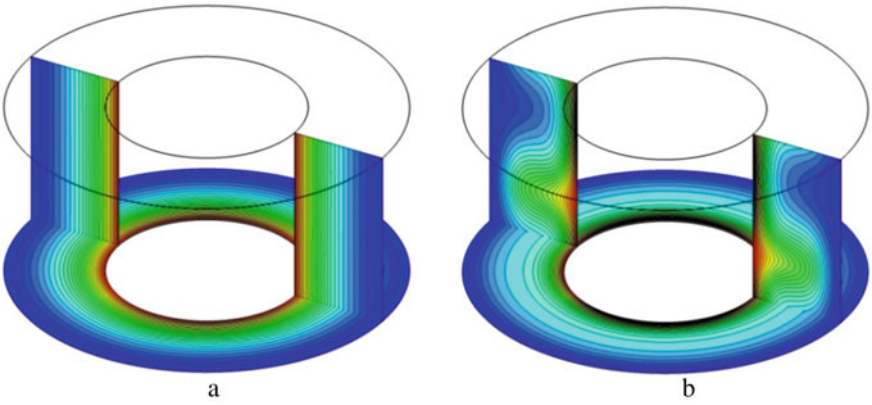


Fig. 5.10 Density distribution: **a** Re = 200, Ω = 0.75, **b** Re = 200, Ω = 0.50

cylinder and near the outer cylinder, the plane vortex flow appears due to the difference in temperature of the cylinders, when the axes of the vortex structures are parallel to the axis of the cylinders. Figure 5.11 demonstrates such flow in the form of the density distribution in the cross-section perpendicular to the axis of the cylinders. There is no movement along the axis of the cylinders in this type of the flow, which is one of the signs of this type of the flow. The flow has a two-dimensional plane character.

Figure 5.11 demonstrates the examples with different numbers of formed vortices (from 8 to 1). Apparently, the maximum number of vortices is limited by the ratio of the length of the average circumference between the cylinders to the distance between the cylinders. At a high speed of the rotation of the outer cylinder, the number of the vortices decreases up to one, while the vortices take a very elongated shape in the circumferential direction. This type of the flow in the considered definition of the problem is connected with the temperature difference between the cylinders, and it can be classified as two-dimensional thermal waves.

The flow when both the Taylor vortices and the plane vortices are formed in the flow is the most interesting one (Figs. 5.12 and 5.13).

In a certain sense, this type of the flow can be considered as the unification of the flows with the Taylor vortices and the plane thermal waves. In this case, the flow is three-dimensional. When considering the flow field in the density distribution, one can observe both the structures elongated in the circumferential direction (corresponding to the Taylor vortices) and the periodic structures in the circumferential direction non-corresponding to the Taylor vortices and corresponding to the plane thermal waves. Rather high values of velocity along the axis of the cylinders appear for this type of the flow (due to the formation of the Taylor vortices, Fig. 5.13a, b).

Figure 5.14 demonstrates the heat flow Q from the outer cylinder to the inner cylinder, depending on the Re number for $\Omega = 0$, $\Omega = 1.0$, and $\Omega = 2.0$ (lines 1, 2, and 3, respectively). The value of Q is related to the temperature difference and the surface area of the inner cylinder (and the coefficient of thermal conductivity of the

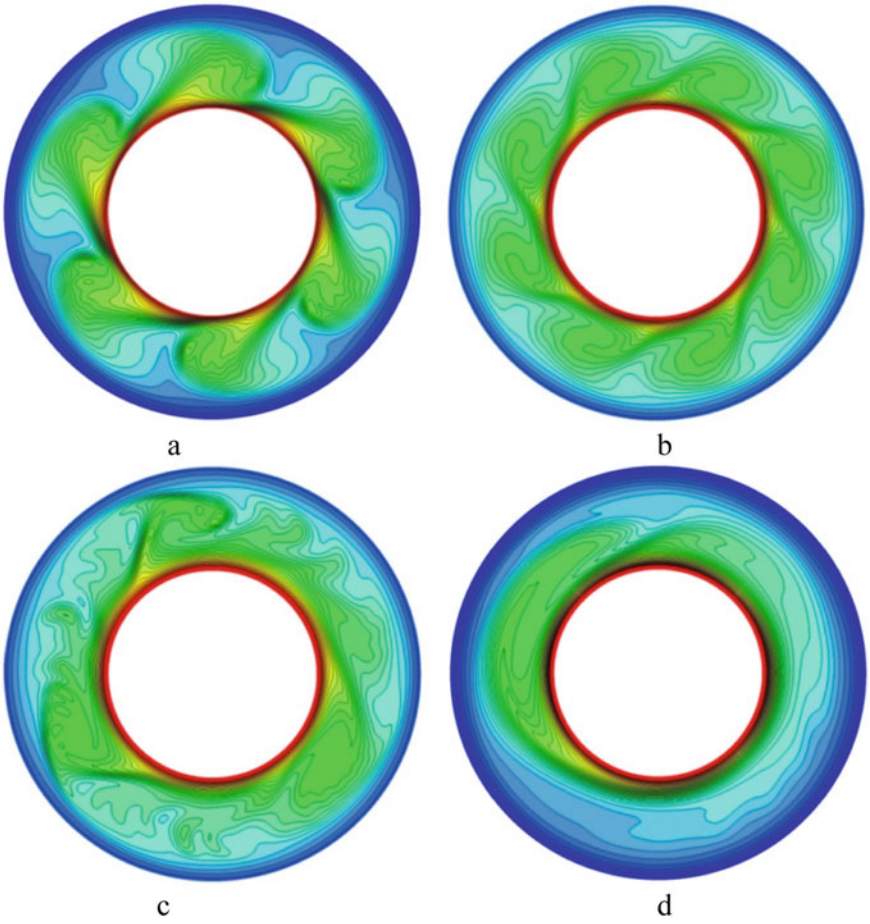


Fig. 5.11 Flow with plane thermal waves: **a** Re = 300, Ω = 1.0, **b** Re = 500, Ω = 1.25, **c** Re = 800, Ω = 1.50, **d** Re = 1000, Ω = 2.0

gas between the cylinders). At Ω = 2.0, Re = 100, heat exchange corresponds to the heat exchange mode for the plane Couette flow. With an increase in the Reynolds number, two-dimensional heat waves At Ω = 1.0, Re = 100, the Couette flow mode is formed, and then the heat exchange is determined at 200 < Re < 500 by plane heat waves. A decrease in the rotation speed of the outer cylinder leads to a decrease in the Re number, at which these waves are formed and in addition slightly increases the heat flow. At Ω = 1.0, Re > 600, as well as at Ω = 0, the presence of the Taylor vortices, whose formation implements the maximum value of Q, influences the heat flow. In this case, the value of the heat flow is almost proportional to the Re number (with the exception of the Re ≈ 100 area, when the Taylor vortices have just formed). At Re = 1000, the heat exchange in the formation of the Taylor vortices in

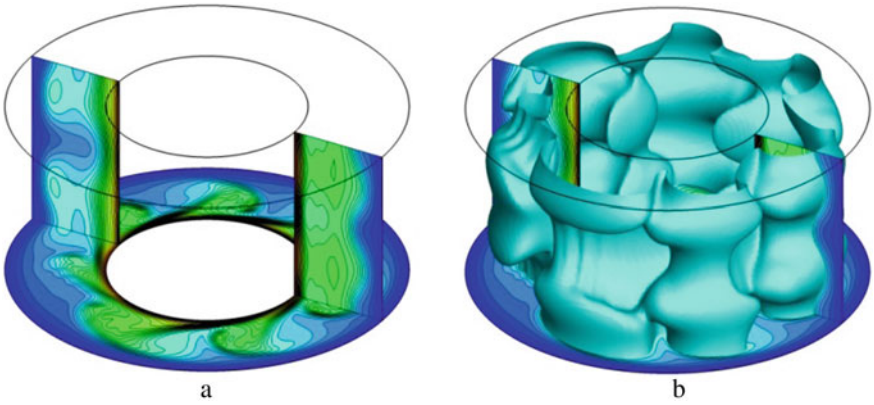


Fig. 5.12 Example of forming the Taylor vortices and the plane vortices, Re = 400, Ω = 1.0: **a** density distribution, **b** constant density surface

comparison with the Couette flow (that is actually a mode without convective heat exchange) is approximately 5 times greater.

Figure 5.15 demonstrates the nature of the heat flow on the surface of the inner cylinder at: Re = 300, Ω = 0.5 (types of flow B—the axisymmetric flow with the Taylor vortices) (Fig. 5.15a), Re = 300, Ω = 1.0 (types of flow C—the plane flow with the vortex structures parallel to the axis of the cylinders) (Fig. 5.15b), and Re = 500, Ω = 1.0 (types of flow D—the three-dimensional flow with the Taylor vortices and the vortex structures parallel to the axis of the cylinders) (Fig. 5.15c). In each case, a different palette range is used.

As the Re number increases, the character of the heat flow from a variable one in the circumferential direction changes to a variable one along the axis of the cylinders. The nature of the flow is the plane waves, the Taylor vortices, or their joint presence that leads to significantly different patterns of heating.

5.5 Conclusions

According to the results of the numerical calculations, the problem of the flow between rotating cylinders with the Taylor vortices allows many solutions, and the same can be seen in the experiments. The choice of the implemented solution can be determined both by additionally imposed conditions (e.g., the finite length of the cylinders) with the corresponding boundary conditions and the history of the establishment, the choice of the initial flow field. The size of the vortices allows a certain range of its value. There exists an optimal value of the Taylor vortex size for maximum friction to appear.

When studying the flow of viscous gas between the cylinders of different temperature, the flow modes with the flat vortex structures and the Taylor vortices, as well

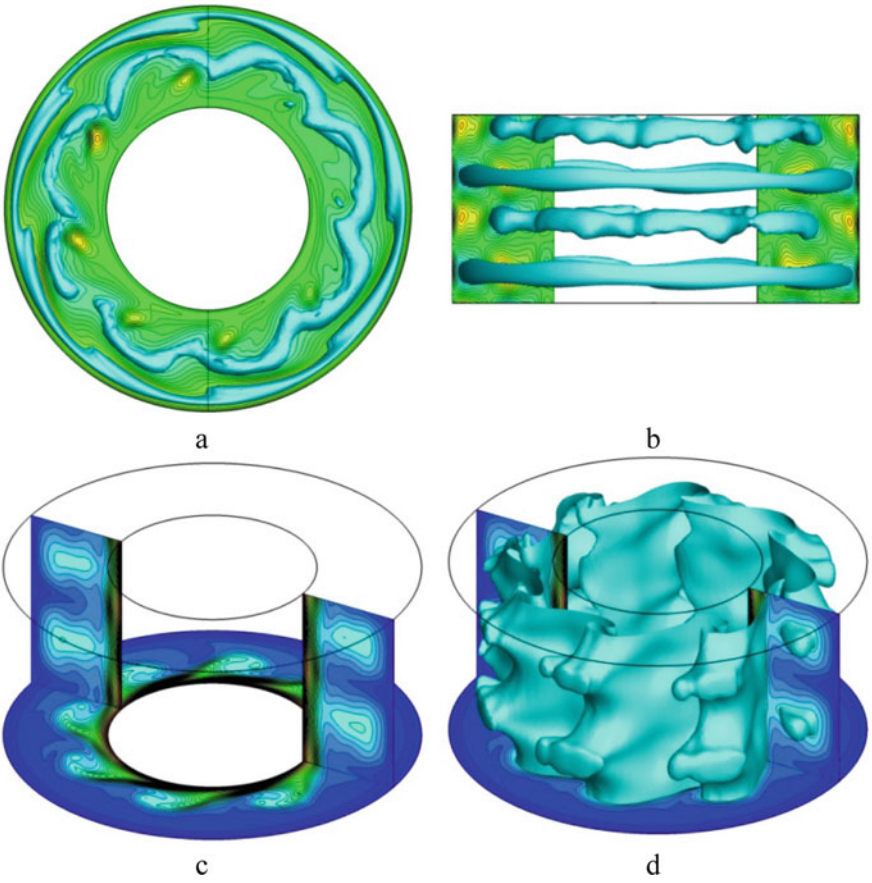


Fig. 5.13 Example of forming the Taylor vortices and the plane vortices, Re = 800, Ω = 1.0, distribution of velocity along the axis of the cylinders **a** axial view, **b** side view, **c** density distribution, **d** constant density surface

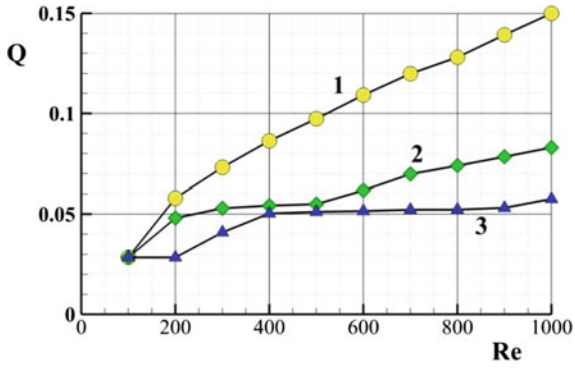


Fig. 5.14 Heat flow depending on the Re number: curve 1 corresponds to Ω = 0.0, curve 2 corresponds to Ω = 1.0, curve 3 corresponds to Ω = 2.0

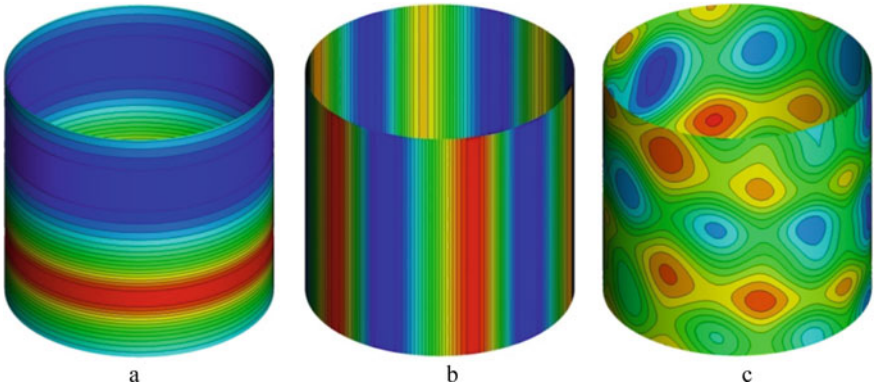


Fig. 5.15 Nature of the heat flow on the internal cylinder: **a** Re = 300, Ω = 0.5, **b** Re = 300, Ω = 1.0, **c** Re = 500, Ω = 1.0

as three-dimensional flow corresponding to the combination of these two types of the flows, were found.

Acknowledgements The calculations were carried out on MVS-100K at Interdepartmental Supercomputer Center of the RAS.

References

1. Taylor, G.I.: Stability of a viscous liquid contained between two rotating cylinders. Philos. Trans. Roy. Soc. **A223**, 289 (1923)
2. Schlichting, H.: Boundary Layer Theory, 4th edn. McGraw Hill, New York (1960)
3. Joseph, D.D.: Stability of Fluid Motions. Springer, Berlin, Heidelberg, New York (1976)
4. Wimmer, M.: Viscous flows and instabilities near rotating bodies. Prog. Aerospace Sci. **25**, 43–103 (1988)
5. Wimmer, M.: An experimental investigation of Taylor vortex flow between conical cylinders. J. Fluid Mech. **292**, 205–227 (1995)
6. Wimmer, M.: Taylor vortices at different geometries. In: Egbers, C., Pfister, G. (eds.) Physics of Rotating Fluids. Lecture Notes in Physics, vol. 549, pp. 194–212. Springer, Berlin Heidelberg (2000)
7. Furukawa, H., Hanaki, M., Watanabe, T.: Influence of initial flow on Taylor vortex flow. J. Fluid Sci. Technol. **3**, 129–136 (2008)
8. Ait-Moussa, N., Poncet, S., Ghezal, A.: Numerical simulations of co- and counter-Taylor–Couette flows: influence of the cavity radius ratio on the appearance of Taylor vortices. Am. J. Fluid Dyn. **5**, 17–22 (2015)
9. Lalaoua, A., Bouabdallah, A.: On the onset of Taylor vortices in finite-length cavity subject to a radial oscillation motion. J. Appl. Fluid Mech. **9**(4), 1887–1896 (2016)
10. Furukawa, H., Suzuki, T.: Study on non-uniqueness of Taylor vortex flow changing inner cylinder acceleration time. World J. Mech. **8**, 301–310 (2018)
11. Maksimov, F.A., Churakov, D.A., Shevelev, Y.D.: Development of mathematical models and numerical methods for aerodynamic design on multiprocessor computers. Comput. Math. Math. Phys. **51**, 284–307 (2011)

12. Kochin, N.E., Kibel′, I.A., Roze, N.V.: In: Radok, J.R.M. (ed.) Theoretical Hydromechanics (translated from the 5th Russian Edition by Boyanovitch, D.). Interscience Publishers, New York (1964)
13. Donnelly, R.J.: Experiments on the stability of viscous flow between rotating cylinders. Proc. Roy. Soc. Ser. A **1246**, 246, 312−325 (1958)

Part II
Numerical Modeling of Plasma and Multiphase Flows

Chapter 6
The Investigation of the Evolution of Cluster Beam Development in the Nozzle-Skimmer System

Igor E. Ivanov, Vladislav S. Nazarov, and Igor A. Kryukov

Abstract The device skimmer is considered, which is intended to separate inert gas clusters with the aim of further collision them with a surface to increase its smoothness order. The condensation and the evaporation processes of Argon in the device are calculated. The modified method of moments (MM) is used for modeling. The droplet nucleation and growth rate coefficients were found by the semi-empirical model. Two-dimensional viscous axisymmetric case is considered. The moment equations are supplemented by the diffusion equation of a condensing gas. To solve the equations, the finite volume method is used. The Riemann problem is solved using the AUSM+ method.

6.1 Introduction

Molecular clusters are widely used to produce new materials [1], deposit thin films on the surfaces [2], and influence surfaces by collision them with a cluster beam [3]. To obtain clusters from gas vapors, it is convenient to use a supersonic gas outflow from a nozzle. The resulting clusters are supposed to be used to bombard the surface of integrated circuits to achieve its greater order of smoothness. However,

I. E. Ivanov · V. S. Nazarov (✉)
Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, Moscow 125993, Russian Federation
e-mail: naz.vladislav@yandex.ru

I. E. Ivanov
e-mail: ivanovmai@gmail.com

I. E. Ivanov
Federal State Budget Educational Institution of Higher Education M.V. Lomonosov Moscow State University (Lomonosov MSU), 1, Leninskie Gory, Moscow 119991, Russian Federation

I. A. Kryukov
Ishlinsky Institute for Problems in Mechanics of the RAS, 01-1, Pr. Vernadskogo, Moscow 119526, Russian Federation
e-mail: ikryukov@gmail.com

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_6

to achieve this goal, it is necessary to separate the clusters using a skimmer. This study examined the nozzle-skimmer system, which is used to obtain Argon clusters. Using bombardment of inert gas clusters in the future, it is planned to get integrated circuits to a higher order of smoothness on an industrial scale.

A cluster is a system consisting of atoms or molecules combined by the forces of van der Waals into a single whole. The number of atoms or molecules in such a system can vary from units to hundreds of thousands [4, 5].

The properties of molecular clusters depend on their size. The properties of small clusters are determined by the structure of the molecules, while the properties of large clusters (droplets) approach the properties of a liquid medium. Clusters are formed as a result of collisions of particles; the following processes are taken into account: elastic collisions of molecules, recombination of molecules, cluster and monomer associations, cluster associations, and monomer evaporation from a cluster [6, 7].

The process of cluster formation begins with a combination of two atoms or molecules (monomers) and obtaining dimers, then trimers, etc. As a rule, small clusters (dimers, trimers, etc.) are obtained as a result of three-particle collisions in which excess energy (the excess energy of colliding particles over the energy of the formed cluster) is carried away by one of the primary particles. A similar three-particle process continues until the formed cluster has a size sufficient to absorb excess energy in the internal degrees of freedom. After this, more often in the process of cluster formation, two-particle reactions are realized [6, 7].

Kinetic models are currently used to study cluster formation. The main method in this direction is statistical modeling using the Monte Carlo method [8–10]. The quasi-chemical model of condensation has become widespread [9, 11, 12]. Simulation of the process of volumetric gas condensation can be carried out based on solving the kinetic equation for the size distribution function of droplets, which describes the evolution of the spectrum of clusters in time and space [13]. However, a direct solution of the kinetic equation is possible only for relatively simple model problems.

The moment method considers two processes that affect the mass fraction of the liquid component: cluster formation (nucleation) and growth of formed clusters due to condensation. Equations describing an evolution of clusters (droplets) in the gas medium flow are obtained as moments from the general equation of dynamics of the size distribution function of cluster droplets. In this work, the system of equations uses the equation for the mass fraction of the liquid phase and the equation for the mass fraction of the condensing phase at the beginning of the calculation (the sum of the mass fractions of the liquid and vapor phases) [14]. Using the equation for the mass fraction of the condensing phase is a new element in the model of moments that allows us to significantly expand the range of problems to be solved, for example, to simulate the outflow of a jet with the condensation of vapor of a condensing substance into a space filled with medium without a condensing substance. Significant development of the method of moments is its generalization to the case of droplet evaporation [15, 16].

In mathematical modeling of the process of homogeneous condensation including moment methods, the results of classical nucleation theory (CNT) are used [16–19].

CNT uses the macroscopic and equilibrium approaches to describe the nonequilibrium process that occurs in most cases on a nanometer scale. Important CNT results are models of nucleation rate, droplet size growth rate, and critical nucleus size.

Recent experimental and computational works suggest that CNT does not adequately describe a nucleation process. Potential sources of errors include: (a) the use of the so-called "capillary approximation" that is the use of the concept of "surface tension" and sphericity for small clusters, (b) the incorrect determination of some macroscopic quantities (densities) in small clusters, (c) not taking into account nonisothermal processes (droplet overheating) during condensation [19–21].

Recently, many approaches to modernizing the classical theory of nucleation have been suggested [22–28]. For example, in [22, 23], the modified relations were proposed for the nucleation rate and droplet size growth rate. These models contain some coefficients that must be selected for each case under consideration. As a reference model for the selection of these coefficients, the authors considered a semi-analytical approximation proposed by Hagena and Obert [29, 30].

Using the method of moments, the chapter investigates the processes of condensation–evaporation during the flow of an argon jet from a micro-nozzle into a vacuum chamber and the argon flow in the nozzle-jet-skimmer system.

Hereinafter, in Sect. 6.2, we consider the mathematical formulation of the model problem. Section 6.3 extends the possibilities of MM due to the Hagena's theory. Section 6.4 contains the results of the numerical modeling. Section 6.5 concludes the chapter.

6.2 Mathematical Model

The propagation of viscous heat-conducting condensing argon vapor in a nozzle-skimmer system is considered. In continuum models, the nucleation function is used to describe the process of cluster formation and further growth of droplets, their growth rate determined from the ratio of partial pressure to vapor pressure. To simulate evaporation, the denucleation function is used in MM instead of the nucleation function [31].

For high concentrations of condensing gas, we can assume that the droplet temperature and medium temperature are the same [32]. In this case, two parameters will have the main effect on condensation: accommodation coefficient α and droplet growth rate β. Each of these parameters can be selected for a specific gas. Previously, the authors considered the case of condensation of water vapor [14, 33] in the experiment [22]. It was shown that the acceptable values of the coefficients are $\alpha = 1$, $\beta = 0.1$.

Section 6.2.1 describes the complete system of equations. Section 6.2.2 realizes method of moments. Section 6.2.3 contains the thermodynamic relations for closing of the equations.

6.2.1 System of Equations

The system of the Navier–Stokes equations supplemented by moment equations and written in a slightly divergent form was taken as a mathematical model of gas:

$$\frac{\partial U}{\partial t} + \frac{\partial (F - F_v)}{\partial x} + \frac{\partial (G - G_v)}{\partial y} = S, \tag{6.1}$$

where

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \\ \rho Q_0 \\ \rho Q_1 \\ \rho Q_2 \\ \rho \alpha \\ \rho \alpha_{\max} \end{bmatrix}, \quad F = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(\rho E + p) \\ \rho u Q_0 \\ \rho u Q_1 \\ \rho u Q_2 \\ \rho u \alpha \\ \rho u \alpha_{\max} \end{bmatrix}, \quad G = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(\rho E + p) \\ \rho v Q_0 \\ \rho v Q_1 \\ \rho v Q_2 \\ \rho v \alpha \\ \rho v \alpha_{\max} \end{bmatrix},$$

$$F_v = \begin{bmatrix} 0 \\ \tau_{yy} \\ \tau_{yx} \\ v\tau_{yy} + u\tau_{yx} - q_x \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad G_v = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{xx} \\ v\tau_{yx} + u\tau_{xx} - q_y \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

$$S = \begin{bmatrix} -\frac{\rho v}{y} \\ -\frac{1}{y}\left(\rho v^2 - \tau_{yy}\right) \\ -\frac{1}{y}(\rho uv - \tau_{xx}) \\ -\frac{v(\rho E + p)}{y} \\ J - \frac{\rho Q_0 v}{y} \\ r_* J + \dot{r}\rho Q_0 - \frac{\rho Q_1 v}{y} \\ r_*^2 J + 2\dot{r}\rho Q_1 - \frac{\rho Q_2 v}{y} \\ \frac{4}{3}\pi\rho_l\left(r_*^3 J + 3\dot{r}\rho Q_2\right) - \frac{\rho Q_3 v}{y} \\ -\frac{\rho \alpha_{\max} v}{y} \end{bmatrix}, \tag{6.2}$$

where

$$\tau_{xx} = \frac{2}{3}\mu\left[2\frac{\partial u}{\partial y} - \frac{1}{y}u - \frac{\partial v}{\partial x}\right], \tag{6.3}$$

$$\tau_{yy} = \frac{2}{3}\mu\left[-\frac{\partial u}{\partial y} - \frac{1}{y}u + 2\frac{\partial v}{\partial x}\right], \tag{6.4}$$

$$\tau_{xy} = \tau_{yx} = \mu\left[\frac{\partial v}{\partial y} + \frac{\partial u}{\partial x}\right], \tag{6.5}$$

$$q_x = -\lambda\frac{\partial T}{\partial x}, \tag{6.6}$$

$$q_y = -\lambda\frac{\partial T}{\partial y}. \tag{6.7}$$

Here, ρ is the density, p is the pressure, T is the static temperature, u is the velocity along the x-direction, v is the velocity along the y-direction, E is the total energy per unit mass, μ is the viscosity coefficient, λ is the thermal conductivity coefficient, J is the nucleation/denucleation function, $\dot{r}$ is the grow rate, Q_0, Q_1, Q_2 are the moments of distribution function.

The system can be considered as a combination of two systems of equations, first of which is the classical system of the Navier–Stokes equations written for a cylindrical coordinate system and the second is a system of moment equations. The equations from Eq. 6.8 to Eq. 6.11 describe the dynamics of the mixture in the two-dimensional representation.

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} = -\frac{\rho v}{y} \tag{6.8}$$

$$\frac{\partial(\rho u)}{\partial t} + \frac{\partial\left(\rho u^2 + p - \tau_{yy}\right)}{\partial x} + \frac{\partial\left(\rho uv - \tau_{xy}\right)}{\partial y} = -\frac{1}{y}\left(\rho v^2 - \tau_{yy}\right) \tag{6.9}$$

$$\frac{\partial(\rho v)}{\partial t} + \frac{\partial\left(\rho uv - \tau_{yx}\right)}{\partial x} + \frac{\partial\left(\rho v^2 + p - \tau_{xx}\right)}{\partial y} = -\frac{1}{y}(\rho uv - \tau_{xx}) \tag{6.10}$$

$$\begin{aligned}\frac{\partial(\rho E)}{\partial t} &+ \frac{\partial\left(u(\rho E + p) - \left(v\tau_{yy} + u\tau_{yx} - q_x\right)\right)}{\partial x} \\ &+ \frac{\partial\left(v(\rho E + p) - \left(v\tau_{yx} + u\tau_{xx} - q_y\right)\right)}{\partial y} = -\frac{v(\rho E + p)}{y}\end{aligned} \tag{6.11}$$

Equations 6.12–6.15 were obtained from the general dynamics equations describing the nucleation process and the dynamics of the homogeneous condensation.

$$\frac{\partial(\rho Q_0)}{\partial t} + \frac{\partial(\rho u Q_0)}{\partial x} + \frac{\partial(\rho v Q_0)}{\partial y} = J - \frac{\rho Q_0 v}{y} \tag{6.12}$$

$$\frac{\partial(\rho Q_1)}{\partial t} + \frac{\partial(\rho u Q_1)}{\partial x} + \frac{\partial(\rho v Q_1)}{\partial y} = r_* J + \dot{r}\rho Q_0 - \frac{\rho Q_1 v}{y} \tag{6.13}$$

$$\frac{\partial(\rho Q_2)}{\partial t} + \frac{\partial(\rho u Q_2)}{\partial x} + \frac{\partial(\rho v Q_2)}{\partial y} = r_*^2 J + 2\dot{r}\rho Q_1 - \frac{\rho Q_2 v}{y} \tag{6.14}$$

$$\frac{\partial(\rho\alpha)}{\partial t} + \frac{\partial(\rho u\alpha)}{\partial x} + \frac{\partial(\rho v\alpha)}{\partial y} = \frac{4}{3}\pi\rho_l\left(r_*^3 J + 3\dot{r}\rho Q_2\right) - \frac{\rho Q_3 v}{y} \tag{6.15}$$

Equation 6.16 describes the propagation of the concentration of the condensing gas.

$$\frac{\partial(\rho\alpha_{\max})}{\partial t} + \frac{\partial(\rho u\alpha_{\max})}{\partial x} + \frac{\partial(\rho v\alpha_{\max})}{\partial y} = -\frac{\rho\alpha_{\max} v}{y} \tag{6.16}$$

6.2.2 Moment Equations

Modeling of condensation in MM occurs through macro-parameters that can be obtained using the first four moments of the distribution function. Increase or decrease in the concentration of the liquid fraction affects a formation of shock waves and changing in the adiabatic coefficient, which completely changes the flow structure. These processes can be considered due to the determination of the concentration of the liquid fraction in the moment equations, the presence of which is taken into account by reconstructing the macro-parameters after solving the system of gas-dynamic equations.

The equations of moments can be represented as an endless chain of moment equations, so called Hill chain [34] described by Eq. 6.17, where $\rho Q_n = \int_{x_*}^{\infty} r^n f(x,t,r)\mathrm{d}r$ is nth order moments.

$$\frac{\partial}{\partial t}(\rho Q_k) + \frac{\partial}{\partial x_i}(\rho U_i Q_k) = (r_*)^k J + k\rho Q_{k-1}\dot{r} \quad k = 1, \infty \tag{6.17}$$

Instead of the moment Q_3, the mass fraction of the liquid fraction $\alpha = 4\pi/3\rho_l Q_3$ is used, where ρ_l is the liquid phase density. In addition, the authors added an equation for $\alpha_{\max}$ to take into account the diffusion of the carrier gas and the vapor of the condensing gas [33].

Nucleation. In MM, two stages of the development of condensation are distinguished. The first stage is nucleation, and the second stage is the growth of the formed clusters. The rate of increase in the number of clusters is determined by the nucleation function J. The dynamics of the growth rate of clusters is transmitted using the growth rate $\dot{r} = \mathrm{d}r/\mathrm{d}t$. Additionally, it is necessary to determine the critical radius r_*, the radius at which droplet growth begins.

Let us use the following relationships [23, 35, 36]:

$$J = \frac{q_c}{(1+\eta)}\sqrt{\frac{2\sigma}{\pi m^3}}\frac{\rho_V^2}{\rho_l}\exp\left(-g\frac{4\pi}{3}\frac{r_*^2\sigma}{R_V m T}\right), \tag{6.18}$$

where $\frac{1}{1+\eta}$ is the corrective factor taking into account the nonstationarity of the process [8], q_c is the condensation coefficient ($q_c \approx 1$), $\eta = 2\frac{\kappa_f - 1}{\kappa_f + 1}\frac{L}{R_V T}\left(\frac{L}{R_V T} - \frac{1}{2}\right)$, $\sigma = k_\sigma \sigma_\infty$, σ_∞ is the flat film surface tension, k_σ is the correction factor taking into account the curvature of the drop, g is the nucleation correction factor multiplier [5],$S = \frac{p_V}{p_S}$ is the saturation parameter,

$$\frac{\mathrm{d}r}{\mathrm{d}t} = \frac{\beta}{\rho_l}\frac{p_V - p_{S,r}}{\sqrt{2\pi R_V T}}, \tag{6.19}$$

where $p_{S,r} = p_S \exp\frac{2\sigma}{\rho_l R_V T r_{Hill}}$ is the saturation pressure on the surface of a drop of average radius size, β is the evaporation coefficient,

$$r_{\text{Hill}} = \begin{cases} \sqrt{\frac{Q_2}{Q_0}} & \text{if } \alpha > 10^{-6} \\ 0 & \text{if } \alpha \le 10^{-6} \end{cases}, \tag{6.20}$$

$$r_* = \begin{cases} \frac{2\sigma}{\rho_l R_V T \ln S} & \text{if S} > 1 \\ \infty & \text{if S} \le 1 \end{cases}, \tag{6.21}$$

where $\rho_V = \rho(\alpha_{\max} - \alpha)$ is the vapor pressure, m is the algebraic notation of a condensing substance. Parameter ρ_l determines the similarly [31].

Evaporation. Evaporation occurs when the vapor pressure becomes less than the saturation pressure. A decrease in the saturation coefficient S is possible at shock waves and as a result of flow deceleration.

The main problem for simulating evaporation is to determine the number of clusters that will evaporate. Usually, for these purposes, the moment distribution function is restored [31, 37]. In [31], only normal and uniform distributions were considered; however, droplets cannot have a negative radius. The lognormal function is devoid of this disadvantage. We made the assumption that the distribution of clusters is lognormal or close for lognormal. In addition, the authors believe that the number of clusters that evaporates occupies a region from 0 to $Q_1/Q_0 - D$ (Fig. 6.1).

Opposite of condensation, there is no need to determine r_* when evaporation occurs. The cluster growth rate is also determined like for condensation, but it will have a negative sign. The denucleation function is determined like a lognormal:

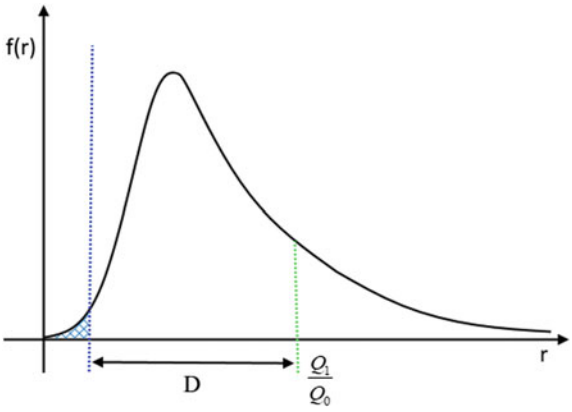


Fig. 6.1 Example of distribution density

$$J = \int_0^{Q_1/Q_0 - D} \varphi(r, m, D)\mathrm{d}r, \tag{6.22}$$

where $\varphi(r, m, D)$ is the density of the lognormal distribution:

$$\varphi(r, m, D) = \frac{1}{r\sqrt{D2\pi}} \exp\left(-\frac{1}{2}\frac{(\ln(r) - m)^2}{D}\right), \tag{6.23}$$

The parameters m, D are obtained from Eq. 6.24 [37].

$$D = \ln\left(\frac{Q_0 Q_2}{Q_1^2}\right) \quad m = \frac{1}{2}\ln\left(\frac{Q_1^4}{Q_0^3 Q_2}\right) \tag{6.24}$$

6.2.3 Thermodynamic Equations

The thermodynamic properties of the mixture are determined by the composition of the components and presence of fluid in it. It is believed that the mixture is in thermodynamic and caloric equilibrium provided by Eqs. 6.25–6.28, where C_{Va}, C_{Pa} are the specific heat components at constant volume and at constant pressure for carrier gas, respectively, C_{VV}, C_{PV} are the same ones for vapor condensing substance, C_{Vmixt}, C_{Pmixt} are the same ones for two-phase mixture, C_l is the specific heat components of liquid, R_a, R_V, R_{mixt} are the individual gas constants of the carrier gas condensing medium and two-phase mixture, respectively, γ_f is the adiabatic exponent of the mixture.

$$C_{\text{Vmixt}} = (1 - \alpha_{\max})C_{\text{Va}} + \alpha_{\max}C_{\text{VV}} + \alpha(C_l - C_{\text{VV}}) \tag{6.25}$$

$$C_{\text{Pmixt}} = (1 - \alpha_{\max})C_{\text{Pa}} + \alpha_{\max}C_{\text{PV}} + \alpha(C_l - C_{\text{PV}}) \tag{6.26}$$

$$R_{\text{mixt}} = (1 - \alpha_{\max})R_a + \alpha_{\max}R_V - \alpha R_V) \tag{6.27}$$

$$\gamma_f = \frac{C_{\text{Pmixt}}}{C_{\text{Vmixt}}} \tag{6.28}$$

The caloric and thermal equations of state are mentioned below in the form of Eqs. 6.29–6.32, where T is the temperature of the mixture, a_f is the frozen velocity of sound of the mixture, L is the latent heat of vaporization.

$$T = \frac{(E - u^2/2) + \alpha L_0}{(1 - \alpha_{\max})C_{\text{Va}} + \alpha_{\max}C_{\text{VV}} + \alpha(C_l - C_{\text{VV}})} \tag{6.29}$$

$$p = \rho T R_{\text{mixt}} \tag{6.30}$$

$$a_f^2 = \gamma_f \frac{p}{\rho} \tag{6.31}$$

$$L = L_1 T + L_0, \quad L_1 = C_{\text{PV}} - C_l \tag{6.32}$$

Viscosity calculation. In the system, a temperature has significant changes. Therefore, it is necessary to take into account the dependence of viscosity on temperature. A modified Sutherland formula uses as the basis for calculating of the viscosity has a view of Eq. 6.33, where $\mu_0 = 1.255 \times 10^{-5}$ kg/m c is the dynamic viscosity for $T_* = 150$ K, $a = 0.945$, $S = 128.35$.

$$\mu = \begin{cases} \mu(T_*)\left(\frac{T}{T_*}\right)^a & \text{if } T < T_* \\ \mu(T_*)\left(\frac{T}{T_*}\right)^{3/2}\frac{T_*+S}{T+S} & \text{if } T \geq T_* \end{cases} \tag{6.33}$$

6.3 The Correction of the Model

In practice, condensation processes have been well studied in the works of Hagena and Obert [29, 30]. They showed that cluster formation and growth depended on pressure p_0, temperature T_0, and nozzle diameter d. Hagena showed that the cluster concentration can be calculated using Eq. 6.34, where Γ^* is the dimensionless similarity parameter of condensation (Hagena parameter).

Table 6.1 Parameters of the Hagena theory

Coefficient of accommodation β	g	n	r_{arg} (m)	α
0.2	2.5	0.23×10^{17}	0.72×10^{-7}	0.1984
0.1	2.0	2.603×10^{17}	0.325×10^{-7}	0.2054
0.1	1.6	5.93×10^{17}	0.251×10^{-7}	0.2143
0.1	1.325	9.096×10^{17}	0.2198×10^{-7}	0.2178

$$N = b\left(\frac{\Gamma^*}{1000}\right) \tag{6.34}$$

For argon, Eqs. 6.35 and 6.36 are applicable, where φ is nozzle expansion angle, $d_* = 750$ mkm, $p_0 = 20,000$ mbar, $T_o = 295$ K, $r_{\text{avrg}} = \left(\frac{Nm}{4/3\pi\rho}\right)$, $m = 6.63 \times 10^{-26}$ kg, $b = 100$, $a = 1.8$.

$$\Gamma^* = 1650 P_0 d_{eq}^{0.85} T_0^{2.29} \tag{6.35}$$

$$d_{eq} = 0.736 d_* / \tan\varphi \tag{6.36}$$

The results of the nozzle under consideration for the Hagena's theory are the following: $N = 7.3 \times 10^5$, $r_{\text{avrg}} = 0.21 \times 10^{-7}$ m, $n = 9.2 \times 10^{17}$ 1/m^3.

Table 6.1 shows the dependence of the parameters obtained at the nozzle exit on the accommodation coefficient and the parameter g.

The best approximation was achieved using the parameter values $\beta = 0.1$, $g = 1.325$.

6.4 Results

The numerical study of the processes occurring in the device of generating cluster beams is carried out. At this stage, the operating modes of the device corresponding to incomplete pumping of gas from the vacuum chamber are considered. The behavior of clusters at the shock wave in front of the skimmer and their further development in the inner region of the skimmer is the subject of interest. The geometry, grid, and boundary conditions of the computational domain are discussed in Sects. 6.4.1–6.4.3, respectively. Section 6.4.4 contains the results of the calculations.

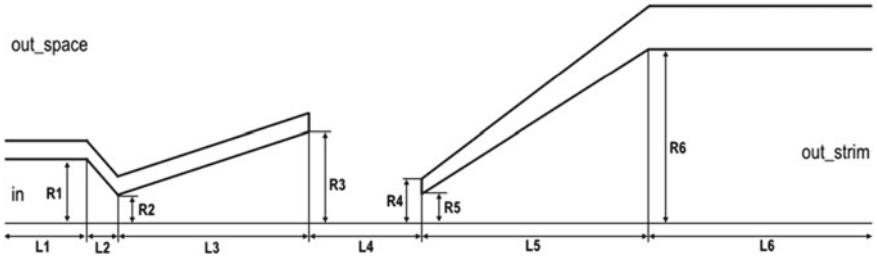


Fig. 6.2 Geometry representation

Table 6.2 Geometry sizes

Index	R_j (m)	L_j (m)
1	0.0007	0.0014
2	0.00007	0.0028
3	0.001418046	0.014
4	0.00046	0.01596
5	0.00025	0.039025
6	0.0125	0.036015

6.4.1 Geometry

The computational domain includes the axisymmetric conical supersonic micro-nozzle, jet flowing out into the vacuum chamber, and axisymmetric skimmer located coaxially with the micro-nozzle at the certain distance from the micro-nozzle cut (Fig. 6.2).

The micro-nozzle starting from the critical section has a conical shape, and the skimmer has the shape of a hollow truncated cone connected to a hollow cylinder. All device sizes are shown in Table 6.2.

6.4.2 Grid

The composite regular computational grid consists of quadrangular cells, is divided into blocks, and has a condensation to the boundaries of the areas coinciding with rigid walls (Fig. 6.3). In total, about 170 thousand design cells (790 $\times$ 210) are present in the computational grid; 6000 cells (200 $\times$ 30) fall on the area inside the micro-nozzle.

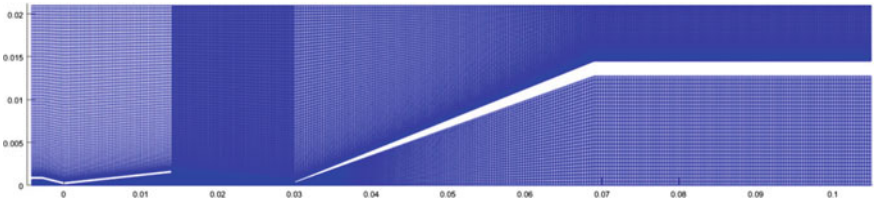


Fig. 6.3 Grid representation

Table 6.3 Boundary conditions

Name	*P* (Pa)	Total temperature (K)
In	709,275 (7 atm)	300
out_space	50	–
out_strim	10	–

6.4.3 Boundary Conditions

For the expansion of the jet to be sufficient for condensation to form, the region in the vacuum chamber behind the nozzle must have very low pressure. The formation of clusters of the required size and their separation occurs in the inner region of the skimmer, where an even higher vacuum is created than in a vacuum chamber. For all boundaries of external areas (out_space), it is assumed that the total pressure on them is the same. The flow parameters at the input and output boundaries are given in Table 6.3.

6.4.4 Calculation Results

The results of calculations of the argon flow taking into account phase transitions (condensation and evaporation) in the micro-system—jet—skimmer system are shown in Figs. 6.4, 6.5, 6.6, 6.7, 6.8, 6.9 and 6.10. In the flow of gas (argon), a boundary layer is formed inside the micro-nozzle, which occupies a significant part

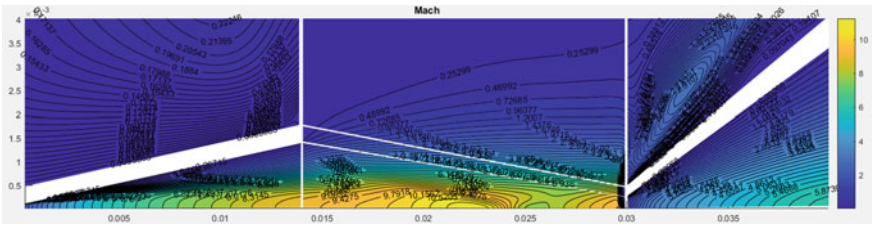


Fig. 6.4. Distribution of the contour of the Mach number in the region of interest

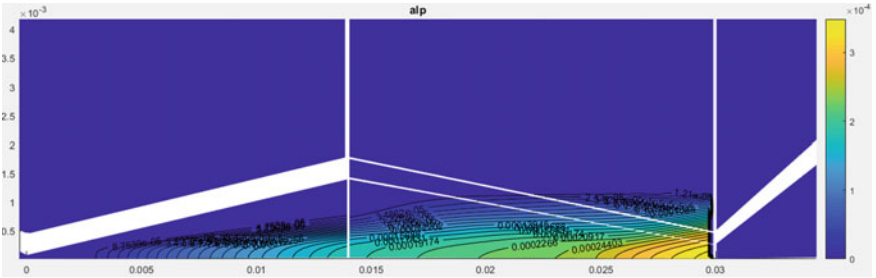


Fig. 6.5. Distribution of contour of the mass fraction of liquid in the region of interest

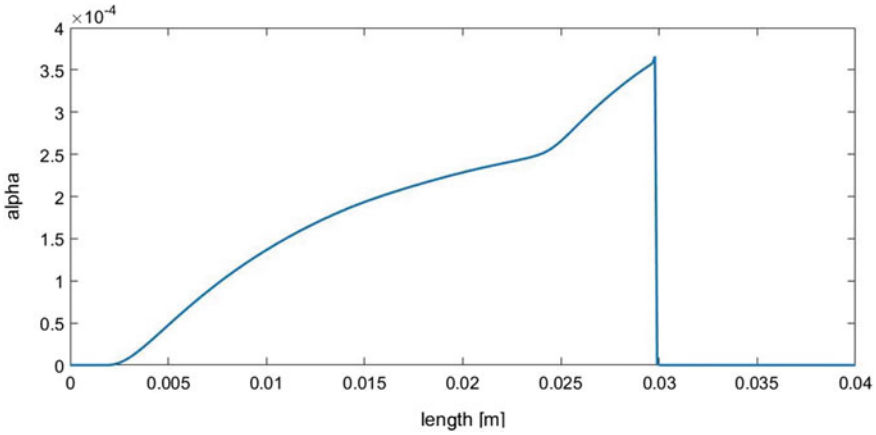


Fig. 6.6. Mass fraction of fluid along the axis of symmetry

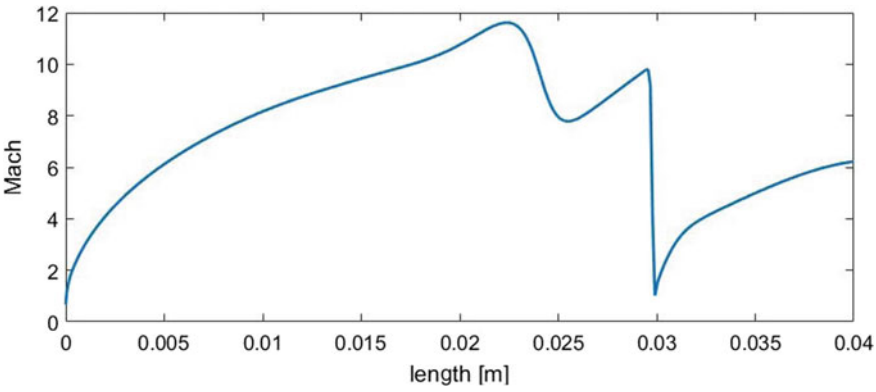


Fig. 6.7. Mach number along the axis of symmetry

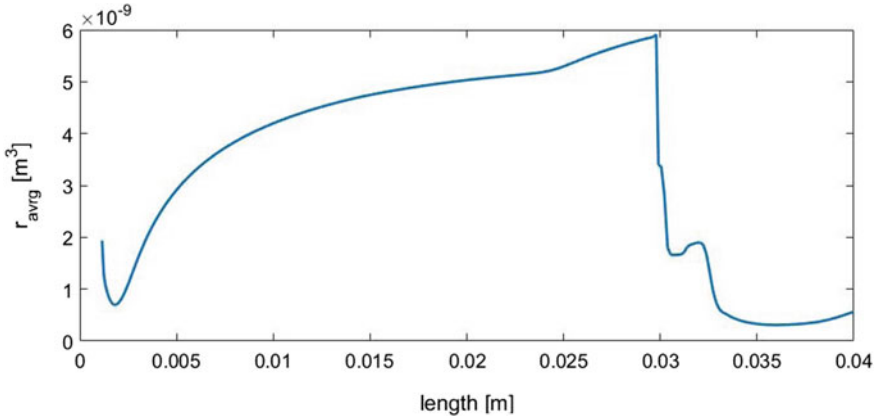


Fig. 6.8 Average radius of the droplet along the axis of symmetry

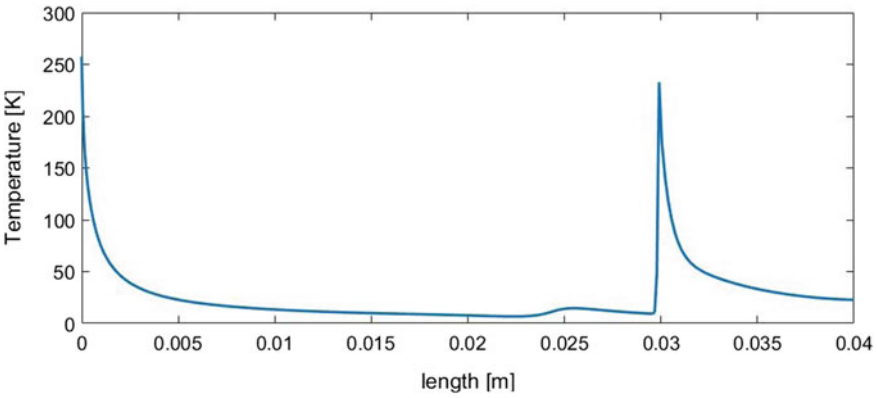


Fig. 6.9 Temperature along the axis of symmetry

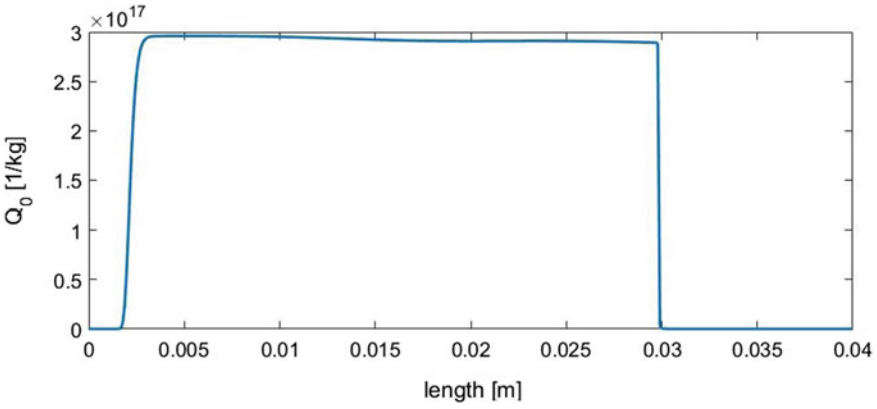


Fig. 6.10 Number of drops along the axis of symmetry

of the inner region of the nozzle (Fig. 6.4). The growth of a thick boundary layer inside the supersonic part of the micro-nozzle sharply reduces the actual degree of expansion of the nozzle. This leads to the decrease in the expansion of the flow in the nozzle and increase in temperature in comparison with the adiabatic inviscid flow. For given parameters (Table 6.3), the jet flowing from the micro-nozzle is under expanded, continues to expand, and accelerate in the open space behind the nozzle. Immediately after the nozzle exit, a standing shock wave arises, which removes excessive over-expansion of the flow in the jet and propagates downstream of the gas. In the region $x = 0.023$–0.025 m, a hanging shock wave falls on the axis of symmetry and is reflected from it in an irregular manner. Due to the large rarefaction of the flow and the insufficiently detailed grid, the structure of irregular reflection with a triple configuration of shock waves is strongly smeared in space and is observed in Figs. 6.7 and 6.9 in the form of a zone of temperature increase and a decrease in the Mach number.

In the calculations, a monotonic increase in the mass fraction of the liquid fraction along the nozzle axis is shown (Figs. 6.5 and 6.6). In this case, behind the zone of irregular reflection of the standing shock wave from the axis of symmetry in the region of the new acceleration (expansion) of the flow, the condensation intensity increases (Figs. 6.5 and 6.6). A strong shock wave arises in front of the skimmer, the front of which is located at $x = 0.029$ m, beyond which the temperature rises sharply (Fig. 6.9). The flow becomes subsonic. In this case, drops of liquid argon behind the front of this shock wave evaporate almost completely (Fig. 6.10) and almost complete denucleation occurs. The mass fraction of liquid droplets, the average radius of the droplet, and the mass concentration of clusters in the flow decrease to almost zero. Subsequently, new nucleation and condensation growth of argon clusters take place in the skimmer.

6.5 Conclusions

The mathematical model of gas-dynamic flows with the phase transformations (condensation and evaporation) is developed. The system of the Navier–Stokes equations is used to describe the flow parameters, and the system of moment equations is used to describe the parameters of a two-phase medium. A numerical algorithm for solving the general system of equations is constructed on the basis of the Godunov scheme with the approximation AUSM+ [38] for solving the Riemann problem.

The developed numerical model was optimized and adapted for the case of pure argon condensation in a nozzle based on the Hagena's semi-empirical theory [29, 30]. In numerical experiments, certain values of the parameters of the condensation model are determined. For example, the values of the coefficient of accommodation and the nucleation correction factor multiplier are determined.

The processes of argon condensation–evaporation in the micro-system-jet-skimmer system for generating cluster beams are studied. The fields of the flow

parameters of two-phase medium and distribution of these parameters along the axis of symmetry are obtained.

Acknowledgements The reported study was funded by RFBR, project number 19-31-90130.

References

1. Khanna, S.N., Linderoth, S.: Magnetic behavior of clusters of ferromagnetic transition metals. Phys. Rev. Lett. **67**(6), 742–745 (1991)
2. Yamada, I., Takaoka, H., Usui, H., Takagi, T.: Low temperature epitaxy by ionized-cluster beam. J. Vac. Sci. Technol. A **4**, 722–727 (1986)
3. Gruber, A., Gspann, J., Hoffmann, H.: Nanostructures produced by cluster beam lithography. Appl. Phys. **68**, 197–201 (1999)
4. Ieshkin, A.E., Kireev, D.S., Ermakov, Yu.A., Trifonov, A.S., Presnov, D.E., Garshev, A.V., Anufriev, Yu.V., Prokhorova, I.G., Krupenin, V.A., Chernysh, V.S.: The quantitative analysis of silicon carbide surface smoothing by Ar and Xe cluster ions. Nucl. Instr. Methods Phys. Res. B **421**, 27–31 (2018)
5. Ieshkin, A., Ermakov, Y., Chernysh, V., Ivanov, I.E., Kryukov, I.A., Alekseev, K., Kargin, N., Insepov, Z.: Computer simulation and visualization of supersonic jet for Gas cluster equipment. Nucl. Instrum. Methods Phys. Res., Sect. A **795**, 395–398 (2015)
6. Karpenko, A.Ju., Baturin, V.A.: Cluster beam sources. Part 1. Methods of cluster beams generation. J. Nano Electron. Phys. **4**, 03015.1–03015.13 (2012) (in Russian)
7. Karpenko, A.Ju., Baturin, V.A.: Cluster beam sources. Part 2. The formation of cluster beams in nozzle sources. J. Nano Electron. Phys. **4**(4), 04015.1–04015.15 (2012) (in Russian)
8. Bykov, N.Y., Gorbachev, Yu.E.: Cluster formation in copper vapor jet expanding into vacuum: the direct simulation Monte Carlo. Vacuum **163**, 119–127 (2019)
9. Bykov, N.Y., Gorbachev, Yu.E.: Mathematical models of water nucleation process for the direct simulation Monte Carlo method. Appl. Math. Comput. **296**, 215–232 (2017)
10. Bykov, N.Y., Safonov, A.I., Leshchev, D.V., Starinskiy, S.V., Bulgakov, A.V.: Gas-jet method of metal film deposition: direct simulation Monte-Carlo of He-Ag mixture flow. Mater. Phys. Mech. **38**, 119–130 (2018)
11. Volkov, V.A., Muslaev, A.V., Pirumov, U.G., Rozovskij, P.V.: Nonequilibrium condensation of metal vapors/inert gas mixture during expansion through the nozzles of cluster-beam generators. Fluid Dyn. **30**, 399–408 (1995)
12. Egorov, B.V., Markachev, Yu.E., Plekhanov, E.A.: Quasi-chemical model of water vapor nucleation. Russ. J. Phys. Chem. B (Khimicheskaya Fizika) **25**(4), 61–70 (2006) (in Russian)
13. Kortsenshtein, N.M., Samuilov, E.V., Yastrebov, A.K.: Study of the volume condensation process in supersaturated vapor by the direct numerical solution of the kinetic equation for the droplet size distribution function. Colloid J. **69**(4), 450–457 (2007)
14. Gidaspov, V.U., Ivanov, I.E., Kryukov, I.A., Nazarov, V.S., Malashin, F.A.: Study of the condensation process in nozzles with a large degree of expansion. Phys.-Chem. Kinet. Gas Dyn. **19**(2), 737.1–737.17 (2018) (in Russian)
15. van Putten, D.S., Sidin, R.S.R., Hagmeijer, R.: Efficient approximation of the cluster size distribution in binary condensation (online no 184511). J. Chem. Phys. **132**(18), 1–9 (2010)
16. Muitjens, M.J.E.H.: Homogeneous condensation in a vapour/gas mixture at high pressures in an expansion cloud chamber. Eindhoven: Technische Universiteit Eindhoven. https://doi.org/10.6100/IR471316 (1996)
17. Becker, R., Döring, W.: Kinetische behandlung der Keimbildung in ubersattingten damfen. Ann. Phys. **24**, 719–752 (1935)

18. Zeldovich, J.: Theory of the formation of a new phase. J. Expl. Theor. Phys. (USSR) **12**, 525 (1942) (in Russian)
19. Frenkel, J.: Kinetic Theory of Liquids. Dover, New York (1955)
20. Kalikmanov, V.I., Wolk, J., Kraska, T.: Argon nucleation: bringing together theory, simulation and experiment. J. Chem. Phys. **128**, 124506.1–124506.8 (2008)
21. Bykov, N.Y., Gorbachev, Yu.E.: On parameters of size-corrected modification of classical nucleation theory for water. AIP Conf. Proc. **1738**, 090008.1–090008.4 (2016)
22. Wyslouzil, B.E., Heath, C.H., Cheung, J.L., Wilemski, G.: Binary condensation in a super-sonic nozzle. J. Chem. Phys. **113**(17), 7317–7329 (2000)
23. Luo, X., Cao, Y., Xie, H., Qin, F.: Moment method for unsteady flows with heterogeneous condensation. Comput. Fluids **146**, 51–58 (2017)
24. Abgaryan, V.K., Gidaspov, V.Y., Nadiradze, A.B., Semenov, A.A.: Ion–electron recombination and heat fluxes in high-frequency ion thrusters. Tech. Phys. Lett. **45**(2), 123–125 (2019)
25. Sova, L., Jun, G., Stastny, M.: Modifications of steam condensation model implemented in commercial solver. AIP Conf. Proc. **1889**, 020039.1–020039.8 (2017)
26. Zhu, X., Lin, Z., Yuan, X., Tejima, T., Niizeki, Y., Shibukawa, N.: Non-equilibrium condensing flow modeling in nozzle and turbine cascade. Int. J. Gas Turbine Propuls. Power Syst. **4**(3), 9–16 (2012)
27. Dykas, S., MajKut, M., Smolka, K., Strozik, M.: An attempt to make, a reliable assessment of the wet steam flow field in the de Laval nozzle. Heat Mass Transf. **54**, 2675–2681 (2018)
28. Bakhtar, F., Young, J.B., White, A.J., Simpson, D.A.: Classical nucleation theory and its application to condensing steam flow calculations. Proc. Inst. Mech. Eng. Part C: J. Mech. Eng. Sci. **219**(12), 1315–1333 (2005)
29. Hagena, O.F.: Cluster beams from nozzle sources: molecular beams and low density gas dynamics. In: Wegener, P.P. (ed.) Molecular Beams and Low Density Gasdynamics, pp. 93–181. Dekker, New York (1974)
30. Hagena, O.F., Obert, W., Chem, J.: Cluster formation in expanding supersonic jets: effect of pressure, temperature, nozzle size, and test gas. J. Chem. Phys. **56**, 1793–1802 (1972)
31. Luo, X.: Unsteady Flows with Phase Transition. Technische Universiteit Eindhoven, Eindhoven (2004)
32. Kortsenshteyn, N.M., Yastrebov, A.K.: Colloid. Droplet temperature distribution in the course of condensation relaxation of supersaturated vapor. Colloid J. **77**(1), 38–45 (2015)
33. Ivanov, I.E., Nazarov, V.S., Gidaspov, V.Yu., Kryukov, I.A.: Numerical simulation of the process of phase transitions in gas-dynamic flows in nozzles and jets. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Theory and Practice of Computational Mechanics: Proceedings of the 21st International Conference on Computational Mechanics and Modern Applied Software Systems, SIST, vol. 173, pp. 133–150. Springer, Singapore (2019)
34. Hill, P.G.: Condensation of water vapour during supersonic expansion in nozzles. J. Fluid Mech. 593–620 (1966)
35. Young, J.B.: The spontaneous condensation in supersonic nozzles. Phys.-Chem. Hydrodyn. **3**(1), 57–82 (1982)
36. Gyarmathy, G.T.: Grundlageiner Theorie der Nassdampfturbine. Dissertation, Juris Verlad, Zurich (1960)
37. Hagmeijer, R., IJzermans, R.H.A., Put, F.: Solution of the general dynamic equation along approximate fluid trajectories generated by the method of moments. Phys. Fluids **17**(5), 056101.1–056101.12 (2005)
38. Nazarov, V.S., Ivanov, I.E., Kryukov, I.A., Gidaspov, V.U.: Modeling the dynamics of a gas-droplet substance in nozzles, taking into account the phase transition. J. Phys.: Conf. Ser. **1250**, 012026.1–012026.10 (2019)

Chapter 7
Numerical Simulation of Generation, Distribution, and Impact of a High-Specific Energy Plasma Bunch on a Barrier

Evgeniy L. Stupitsky, Andrey A. Motorin, and Darya S. Moiseeva

Abstract Physical and comprehensive numerical studies of the generation of plasma bunches with a high specific energy have been carried out with the use of a plasma gun. The parameters of the plasma bunch upon exit from the plasma accelerator and during propagation in the ionosphere ($h > 200$ km) to considerable distances ($\approx$100 km) have been calculated. A special numerical algorithm is presented to determine the results of the impact of a rarefied high-velocity gas flow ($\sim 5 \times 10^7$ cm/s) on the surface of crystalline and amorphous solid bodies. Based on the results, the electron concentration and the scale of the ionized region that formed during the passage of a high-speed toroidal plasma bunch through the rarefied air were estimated. When the bunch spreads at a height $\sim$120 km and a distance $\sim$50 km, the ionized area with transverse dimensions of $\sim$20 km has an electron concentration of $\sim 6 \times 10^8$ cm^{-3}.

7.1 Introduction

At present, studies on the design and use of plasma guans are in active development. In Russia, a large cycle of experimental works were carried out in TRINITI on both the development of generators of plasma bunches with a high specific energy and the study of their pulsed action on a solid body target. In the USA, there was also a

The work was performed within the state task of the ICAD RAS.

E. L. Stupitsky · A. A. Motorin (✉)
Institute for Computer Aided Design of the RAS, 19/18, Vtoraya Brestskaya ul, Moscow 123056, Russian Federation
e-mail: vansp91@gmail.com

E. L. Stupitsky
e-mail: stup@bk.ru

E. L. Stupitsky · D. S. Moiseeva
Moscow Institute of Physics and Technology (National Research University), 9, Institutsky Per., Dolgoprudny, Moscow 141701, Russian Federation
e-mail: moiseevads@rambler.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_7

series of works to create a pulsed plasma generator with highest energy and dynamic characteristics [1]. A sufficiently informative and, at the same time, brief survey of works in the leading plasma laboratories of the USA in this area was made in [1]. The main experimental results of that work were on studies of the MARAUDER facility. Their goals were an analysis of the generation of a Compact plasma Toroid (CT) with a high specific energy and estimation of the characteristics of the high-temperature plasma region that forms when the incident flow interacts with the flow reflected from the solid body target directly at the CT exit from the generator, the transverse and longitudinal scales of which are on the order of meters.

The distinctive and practically important feature of the works of TRINITY is, first of all, that the scales of the generator are considerably smaller, which can make its application domain rather wide. When a plasma bunch moves in vacuum or strongly rarefied gas, its scales increase; the density, temperature, and ionization degree decrease. Therefore, the character of the interaction with the surrounding rarefied gas medium and geomagnetic field changes. The study of these processes is important for a series of fundamental and applied problems of plasma physics. Since the scales of the bunch in this process significantly increase and it ceases to be compact, we call it the Toroidal Plasma Bunch (TPB).

Based on a two-dimensional numerical algorithm developed earlier [2], the very initial stage of the dynamics of TPB upon exit from the generator and motion in vacuum was studied, as well as, the interaction of the incident flow with the flow reflected from the barrier; the parameters of the electromagnetic disturbance generated in this process were determined.

In connection with the creation of small-size plasma generators, the domain of their practical use, certainly, expands significantly, primarily due to the possibility of bunch propagation to considerable distances in a rarefied atmosphere. Thus, such devices are often called plasma guns.

The operation of space technique in the Near-Earth Space (NES) is based to a considerable extent on the use of OptoElectronic Equipment (OEE). Both in OEE itself and in the protection tools, glass-type dielectric materials with a high degree of transparence are used.

Thus, analysis of the possible use of plasma guns for studies of scientific and applied questions of the dynamics of plasma bunches in the ionosphere includes the successive solution of three main problems:

- Formation of TPB and its motion at the initial stage just after the exit from the generator, when the decisive influence on its structure is caused by the internal toroidal current and the poloidal magnetic field generated by it.
- Numerical study of the motion of partially ionized TPB in a rarefied atmosphere, when the internal current is already small and the decisive influence on its structure and motion is caused by processes of the collisional interaction of the bunch plasma with the rarefied surrounding gas when they mutually penetrate each other due to the high-velocity relative motion.
- Numerical study of the impact of the rarefied gas flow formed by TPB on the surface of the transparent dielectric. The goal of the work is to mathematically

model the initial and subsequent stage of TPB motion in a rarefied gas and the impact of a high-velocity flow on the surface of a solid body.

The chapter is organized as follows. Section 7.2 describes the general principles of plasma gun functioning and the physical picture of plasma clot formation. Section 7.3 provides a physical and mathematical statement of the initial stage of the problem to be solved. Due to the fact that this problem is quite complex both mathematically and especially physically, it made sense to solve the problem sequentially numerically with gradual connection of physical processes. Section 7.4 presents the results of the calculation in the adiabatic approximation (without taking into consideration internal kinetic processes and Joule heating). And in Sect. 7.5, calculations of the initial stage of CT dynamics are given in full statement. Based on studies of the initial stage of formation and movement of CT, calculations of its further movement in the ionosphere were made and its influence on the ionosphere was estimated. Section 7.6 is dedicated to this. Section 7.7 shows the effect of a rarefied plasma flow of certain duration on a crystal and amorphous structure of the glass type. Section 7.8 concludes the chapter.

7.2 Physical Picture of the Formation and TPB Dynamics

Electromagnetic shock tubes used for the creation of intense shock waves have been known for a rather long time. Their operation is based on the effects of gas heating by an electric discharge and its acceleration under the action of magnetic forces. The discharge current flows in the radial direction between electrodes, one of which is a rod positioned on the tube axis and the other is a cylinder near the tube surface. The radial current of the discharge interacts with the concentric magnetic field of the current itself and the current flowing along the central electrode. The ponderomotive force is directed along the tube axis and accelerates the plasma in this direction. The further development of works in this area was related to the creation of plasma guns. An important distinctive feature of plasma guns is the presence of a powerful toroidal current in the torus-shaped plasma bunch ejected from the generator. As shown by [1], the radial magnetic field forming the toroidal current is created by a special current coil placed in the initial part of the channel. During CT motion in this field, the toroidal current creates its own poloidal magnetic field, which exercises a decisive influence on the preservation of the compact structure of the plasma bunch at the initial stage of its motion after the escape from the generator.

The geometry of the plasma bunch upon exit from the pulsed plasma accelerator has approximately the following characteristics, as shown in Fig. 7.1.

We assume that the toroidal current in TPB is determined by the radial component of the magnetic field B_r. The component is created by a coil with a current of $I_k \cong 5$ kA and a distance of $\delta \cong 1-3$ cm from the coil to TPB surface. The radial component of the magnetic field in the region inside TPB can then be estimated as

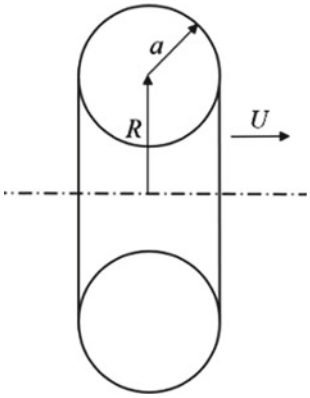


Fig. 7.1 Geometry of TPB. The initial conditions are the following: $a_0 = 3.5$, $R_0 = 6.5$ cm. Hydrogen plasma with a bunch mass of $M = 2.2$ mg

$$B_r = \frac{1}{2a_0}\int_{\delta}^{2a_0}\frac{\mu_0 I_k}{2\pi r}\mathrm{d}r = \frac{\mu_0 I_k}{4\pi a_0}\ln\left(\frac{2a_0}{\delta}\right), \tag{7.1}$$

where $\ln\left(\frac{2a_0}{\delta}\right) \approx 1$, μ_0 is the magnetic constant. Since hydrogen is completely ionized inside the torus during the motion in the channel, then for $T = 10\,\mathrm{eV}$, $n_i = 8.3\times10^{17}\,\mathrm{cm}^{-3}$, the electron–ion energy exchange time is obtained $\tau_{\mathrm{ei}} = 1.9\times10^{-9}$ s and the conductivity is equal to $\sigma = 4.87\times10^7$ S/m (here S is Siemens).

The toroidal current can be estimated as $I = S\sigma U B_r$. Setting area of the transverse cross section $S = \pi a_0^2 = 3.85\times10^{-3}\,\mathrm{m}^2$, $U = 3\times10^5$ m/s, and $B_r = 0.013$ T, we obtain $I \cong 0.7$ MA. Such current creates a poloidal magnetic field $B_\varphi \approx \mu_0 I/2\pi a_0 \cong 4$ T on the toroidal surface.

Therefore, the initial energy distribution (kJ) in TPB upon exit from the generator is approximately as follows:

$$E_k = \frac{MU^2}{2} = 100,$$

$$E_T = \frac{3}{2}kT2N = 0.63T_{\mathrm{eV}} = 6.3 - 18.9,$$

$$E_i = \mathcal{T}N = 2.85.$$

The ionization energy E_i and thermal energy E_T are considerably less than the kinetic energy E_k of the directed motion, where the ionization potential of hydrogen $\mathcal{T} = 13.6\,\mathrm{eV}$ and N is total number of hydrogen ions.

According to the performed analysis, the initial TPB stage after exit from the generator, when the magnetic field created by the toroidal current exercises a decisive influence on its parameters, can be calculated with the following sufficiently matched

initial parameters: $I_0 = 0.7\,\text{MA}$, $a_0 = 3.5\,\text{cm}$, $R_0 = 6.5\,\text{cm}$, $T_{e0} = T_{i0} = 10\,\text{eV}$, $n_{i0} = 8.3 \times 10^{17}\,\text{cm}^{-3}$, and $U_0 = 3 \times 10^7\,\text{cm/s}$.

For a more comprehensive understanding of TPB evolution during motion, the posed problem in this work was gradually detailed and refined. At first, the behavior of a homogeneous cylindrical layer in a one-dimensional cylindrical approximation is considered. The gas is supposed to be ideal.

The calculation was carried out with Brode's difference scheme [3] and von Neumann artificial viscosity. As follows from the dimensionless equations and initial conditions, the only parameter determining flow development is $\overline{a}_0$. The calculations showed that at a zero value of the initial velocity $\overline{u} = u/u_x$, where $u_x = \sqrt{\gamma k T_0/m}$. The ring-shaped structure quickly turns into an expanding solid disk in the absence of any external forces and field inside the toroid for any initial $\overline{a}_0$, on which the collapse time depends. However, as calculated studies have shown, if $\overline{u}(\overline{t} = 0) \geq 2$, then the internal rarefaction wave does not reach the center and the annular expansion will continue in time. It should be noted, that this value $\overline{u} = 0$ is less than the maximum value $\overline{u} = 2(\gamma - 1) = 3$, which according to the front of the plane rarefaction wave, since in this case the movement is cylindrical and not self-similar.

7.3 Physico-mathematical Formulation of the Problem of the Initial Stage of TPB Dynamics

TPB dynamics is described by variation in the two main parameters, $R(t)$ and $a(t)$. The variation $R(t)$ is determined by the radial tension created by the pressure of the magnetic field inside the toroidal ring and is proportional to the squared current I^2 flowing inside the torus, as well as, by the action of the internal pressure inside TPB. The expressions for both components of the force were obtained in studies [4]. Using them, one can approximately write the equation for $R(t)$ in the form of Eq. 7.2.

$$M_0 \frac{\mathrm{d}^2 R}{\mathrm{d}t^2} = 2\pi^2 a^2 \overline{P} + \frac{I^2}{2c^2} \frac{\partial L}{\partial R} \tag{7.2}$$

Here, $\overline{P} = nkT + n_e k T_e$ is the average plasma pressure inside TPB, and $L = 2\pi R\left[\ln\left(\frac{R}{a}\right) + 0.25\right]$ is the torus inductance, c is the speed of light.

To calculate the variation $a(t)$, we disregard the difference of pressures at the inner and outer envelopes of the torus boundary and use the equation for the plasma cylinder with the longitudinal current. For the velocity inside it, we have Eq. 7.3.

$$\rho \frac{\mathrm{d}U}{\mathrm{d}t} = -\frac{\partial P}{\partial r} + \frac{1}{c}[\mathbf{j} \times \mathbf{B}]_z \tag{7.3}$$

Taking into account that $U(t, r = 0) \cong 0$ and the fact that the magnetic pressure at the boundary decreases with time, we approximately assume that the velocity varies linearly along the radius:

$$U(r,t) = r\frac{\dot{a}(t)}{a(t)}. \tag{7.4}$$

From Maxwell's equations, we have:

$$\vec{\mathbf{j}} = \frac{c}{4\pi}\mathrm{rot}(\vec{\mathbf{B}}). \tag{7.5}$$

Given that the magnetic force is associated only with the presence of a poloidal field $B_\theta = B$, $j_z = j$, it follows from Eq. 7.2 [5]:

$$4\pi\rho r^3\frac{\ddot{a}}{a} = -4\pi r^2\frac{\partial P}{\partial r} - B_r\frac{\partial B_r}{\partial r}.$$

Assuming homogeneous distribution of all parameters (except field B) over radius a and integrating over $0 \le r \le a$, in [4], a closed system of equations was obtained for determining the dynamic parameters of a torus: the inner $a(t)$ and outer $R(t)$ radii of the cylindrical layer, velocities of the toroidal ring $U_R(t)$, and toroidal cross section $U_a(t)$ and current $I(t)$:

$$M(t)\frac{\mathrm{d}U_a}{\mathrm{d}t} = 4\pi a P - \frac{2I^2}{c^2 a}, \tag{7.6}$$

$$M_0\frac{\mathrm{d}U_R}{\mathrm{d}t} = 2\pi^2 a^2 P + \frac{\pi I^2}{c^2}\left[\ln\frac{R}{a} + \frac{5}{4}\right], \tag{7.7}$$

$$\frac{\mathrm{d}a}{\mathrm{d}t} = U_a, \tag{7.8}$$

$$\frac{\mathrm{d}R}{\mathrm{d}t} = U_R, \tag{7.9}$$

$$\frac{\mathrm{d}I}{\mathrm{d}t} = -\frac{I}{\tau}, \tag{7.10}$$

where

$$M(t) = M_0/2\pi R, \quad \tau = \frac{L}{c^2 R_c + \dot{L}}, \quad \dot{L} = \mathrm{d}L/\mathrm{d}t, \tag{7.11}$$

$M(t)$ is the mass of the unit length of the toroidal ring.

The pressure is defined by Eq. 7.12, where a is the ionization degree and n is the density of heavy particles.

$$P = kn(T_i + \alpha T_e) = \overline{P} \tag{7.12}$$

L is the torus inductance and can be estimated using Eq. 7.13.

$$L = 2\pi R\left[\ln\frac{R}{a} + \frac{1}{4}\right] \tag{7.13}$$

The resistance of the toroidal ring is determined by Eqs. 7.14 and 7.15, where m is the mass of the electron, n_e is the density of electrons.

$$R_c = \frac{1}{\sigma}\frac{2\pi R}{\pi a^2} \tag{7.14}$$

$$\sigma = \frac{e^2 n_e}{m_e \nu_e} \tag{7.15}$$

To determine $P(t)$ and $\sigma(t)$, it is necessary to know the time behavior of volume-averaged TPB values $T_e(t)$, $T(t)$, and $\alpha(t)$. The collision frequency ν_e entering into $\sigma(t)$ in the general case of the plasma charge composition is determined by following expressions:

$$\nu_e = \nu_{e0} + \nu_{ei},$$

$$\nu_{e0} = \frac{4}{3}\sigma_{e0}\overline{V}_e n_0, \quad \overline{V}_e = \sqrt{\frac{8kT_e}{\pi m_e}}, \tag{7.16}$$

$$\nu_{ei} = \frac{4\sqrt{2\pi}e^4 L_k}{3\sqrt{m_e}(kT_e)^{3/2}}\sum z^2 n_z. \tag{7.17}$$

For the electron concentration n_e, excited particles and temperatures, Eqs. 7.18–7.21 were used. Thus, the kinetics equations for the relative densities of electrons α and excited atoms α_1 have the form of Eqs. 7.18 and 7.19.

$$n\frac{\mathrm{d}\alpha}{\mathrm{d}t} = \left(n_0 n_e - n_e^3 j_{ei} - n_e^2 j_{ei}^{v}\right) \tag{7.18}$$

$$n\frac{\mathrm{d}\alpha_1}{\mathrm{d}t} = (n_e n_0 j_{01} - n_e n_1 j_{10}) - \left(n_e n_1 j_{1e} - n_e^3 j_{e1} - n_e^2 j_{e1}^{v}\right) - A_{10} n_1 \tag{7.19}$$

If the expansion occurs rapidly, the temperature of heavy particles T can move away from the electron temperature T_e. For this reason, the problem was considered in the two-temperature approximation provided by Eqs. 7.20 and 7.21 where j_{ei}, j_{01}, j_{10}, j_{ei}^{v}, j_{e1}, j_{e1}^{v}, j_{1e}, A_{10}, E_{01}, F, J are the constants of process rates [4], Q_{0e}, Q_{ie} describes the rate of energy transfer from electrons to neutral particles and ions.

$$\frac{3}{2}n_e\frac{\mathrm{d}T_e}{\mathrm{d}t}+n_eT_e\mathrm{div}\,\overrightarrow{U}=-\left(J+\frac{3}{2}T_e\right)\cdot n_e j_{ei}\cdot\left(n_0K-n_e^2\right)-E_{01}n_e(n_0j_{01}-n_1j_{10})$$
$$+T_e\left(\frac{3}{2}-F\right)n_e^2j_{ei}^{v}-S_e^{v}+Q_{e0}+Q_{ei}+Q_g \tag{7.20}$$

$$\frac{3}{2}n\frac{\mathrm{d}T}{\mathrm{d}t}+nT\,\mathrm{div}\,\overrightarrow{U}=Q_{0e}+Q_{ie} \tag{7.21}$$

From the equation of continuity $\frac{\partial n}{\partial t}+n\mathrm{div}\mathbf{U}=0$ for the term describing the adiabatic expansion, we obtain Eq. 7.22 where $V=2\pi R\pi a^2$ is the torus volume.

$$\mathrm{div}\mathbf{U}=\frac{1}{V}\frac{\mathrm{d}V}{\mathrm{d}t} \tag{7.22}$$

All designations are generally accepted. Expressions for the rate constants and energy exchange are given in [6]. For Joule heating of electrons, the approximate expression can be obtained $Q_g=I^2/\sigma\pi^2a^4k$.

To solve the most complex equations for $T_e(t)$ and $T(t)$, the splitting method for physical processes was used: $y_e=T_e/T_g,\quad y_i=T/T_g$, where T_g is determined by adiabatic expansion $T_gV^{2/3}=\mathrm{const}$. At that for y_e and y_i, ordinary differential equations are obtained for [5]. To get an idea of the interaction of the geometric characteristics with the current and the magnetic field, the problem was first solved under the assumption: $T_e=T_i=T,\ y_e=y_i=1,\ \alpha=\alpha_0$, i.e., practically in the adiabatic approximation.

Since the initial determining parameters are current I_0 and temperature T_0, then the quantity $\beta(t=0)=\frac{\overline{P}_0}{B_0^2/8\pi}=0.0204\,\frac{T_0}{I_0^2}\left(\frac{\mathrm{eV}}{\mathrm{MA}}\right)$ depends on them.

7.4 Numerical Results and Their Analysis

Calculations performed in the adiabatic approximation demonstrate that the behavior of $a(t), U_a(t), T(t)$, and $n(t)$ significantly depends on the relation between the thermal pressure inside the torus and magnetic pressure created by the current inside the torus, i.e., on $\beta(t=0)$. The calculations were performed for different T_0 and I_0 to implement different values of $\beta(t=0)$ and, at the same time, to keep them within possible experimental values. Figure 7.2 shows some values of $\beta(t=0)$. One should emphasize that main initial parameters determining TPB behavior after leaving the generator—T_0 and I_0—are not related to each other directly because T_0 is determined by the discharge power and I_0 is determined primarily by the presence of the radial component of the magnetic field created in the generator in some way. Thus, the effect of T_0 and I_0 on the change in TPB characteristics after leaving the generator was first studied.

Various calculation options were obtained with varying values β. The results are presented in Fig. 7.2. The behavior of the main parameters at an early stage

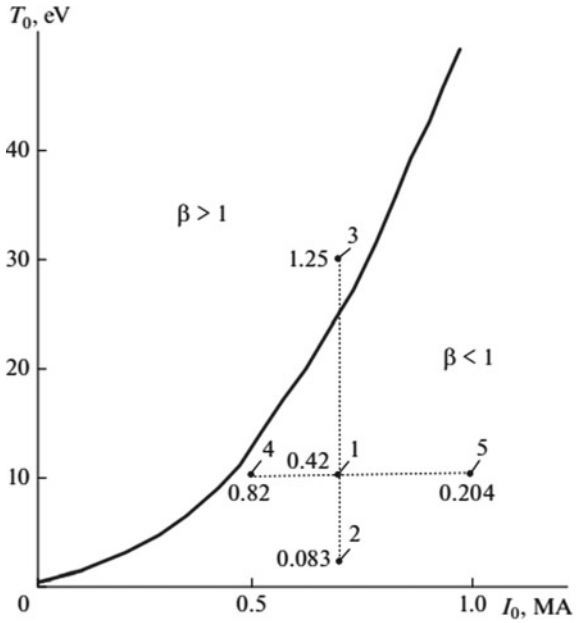


Fig. 7.2 Dependence $\beta(t = 0) = 1$ in the plane T_0, I_0. Variants: (1) $I_0 = 0.7$ MA, $T_0 = 10$ eV, (2) $I_0 = 0.7$ MA, $T_0 = 2$ eV, (3) $I_0 = 0.7$ MA, $T_0 = 30$ eV, (4) $I_0 = 0.5$ MA, $T_0 = 10$ eV, (5) $I_0 = 1$ A, $T_0 = 10$ eV

of expansion is shown in Fig. 7.3. Two new physical effects should be noted. The monotonic increase in $R(t)$ is caused by the fact that both thermal and magnetic pressure are directed along the radius from the torus center. For $a(t)$, the magnetic and thermal pressures are directed oppositely, which causes oscillations of $a(t)$ and $U_a(t)$ because $\beta(t = 0) \neq 1$, and the initial state is dynamically nonequilibrium. Oscillations do not appear at $\beta(t = 0)$ close to unity (variants 3 and 4). Since the oscillations appear in the absence of an external periodic action and their character is determined solely by the system itself, they can be treated as a certain class of nonlinear self-oscillations.

The variation in T_0 at given I_0 (for definiteness, it was taken that $I_0 = 0.7$ MA) demonstrates that if $T_0 < 1.95$ eV, then the velocity U_a very rapidly exceeds U_R and the torus collapses to a solid disk, as it occurs similarly in the model problem at small initial values of its radial expansion. The fundamental difference is that the radial expansion rate in this case is formed not only by the thermal pressure but also by the magnetic field. At $T_0 > 1.95$ eV and $I_0 = 0.7$ MA, the magnetic pressure makes it possible to sustain the toroidal structure, as follows from the considered adiabatic approximation, at least at the initial stage of TPB dynamics.

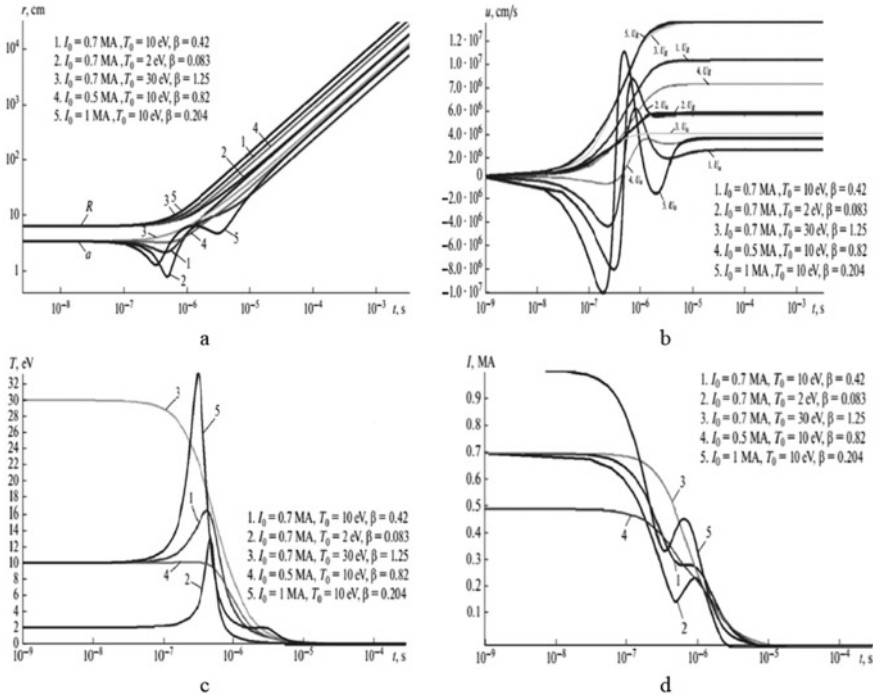


Fig. 7.3 Behavior of the main dynamic parameters at an early stage of expansion: **a** radii of TPB, **b** velocities of radius change, **c** temperature, **d** current

7.5 Calculations of the Initial Stage of TPB Dynamics in Full Statement

In the full problem under consideration, one can distinguish the following main characteristic times:

- Time of current relaxation ($\beta \to \infty$).
- Relaxation time of the ionization and temperature of the nonequilibrium state.
- Time to reach the full, mutual penetration of air particles through TPB during its directed motion.
- Time of TPB deceleration in a rarefied atmosphere.

As shown by calculations (Fig. 7.4), the relative contribution of ionization, excitation, and Joule heating depends on T_0, I_0. It follows from Fig. 7.4 that the largest positive contribution to T_e in the region of largest time gradients is made by the Joule heating and recombination. The role of radiation is insignificant.

In general, the behavior of the geometrical parameters R, a, U_R, U_a, temperature T, and current I is similar in the form to results obtained in the adiabatic approximation. Due to the Joule and recombination energies, the asymptotic values $U_{R\infty}$ and

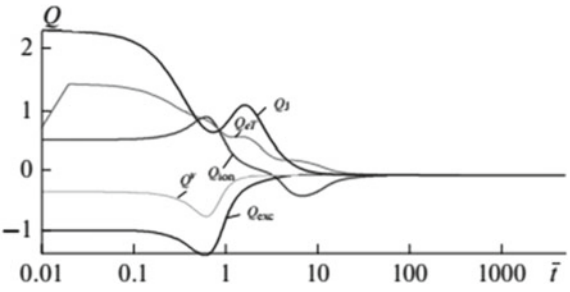


Fig. 7.4 Contribution of different processes to the internal energy at the initial stage of its dynamics for different initial conditions

$U_{a\infty}$ are approximately larger (20%) than in the adiabatic case. The main decrease in the current temperature occurs somewhat longer ($\sim 3 \times 10^{-5}$ s).

However, the full formulation made it possible to clarify two important effects. Since the particle density rapidly drops with time, the recombination rate decreases and the implemented regime is close to quenching of the ionization degree with sufficiently large asymptotic values α_∞(70–75%), see Fig. 7.5. Therefore, the conductivity of the bunch plasma remains high during further TPB motion, which determines its interaction with the geomagnetic field. However, as shown by Stupitskii et al. [5] complete quenching does not occur. In these calculations, this is indicated by the increase in kinetic temperatures y_e and y by $t \geq 10^{-5}$ s which decelerates the drop of T_e and T at a later stage, at $t \geq 3 \times 10^{-4}$s, the collisions no longer provide temperature equalization, and $T_e > T$, although the excesses are insignificant (Fig. 7.6).

Note that the decrease in the initial TPB energy, a decrease in T_0 or I_0, leads after a certain period to the collapse of the torus to a disk, as mentioned in the adiabatic approximation. Calculations in the full formulation demonstrate that the torus collapses to a solid disk at $T_{0\text{cr}} \approx 0.6$ eV in the whole range of the considered

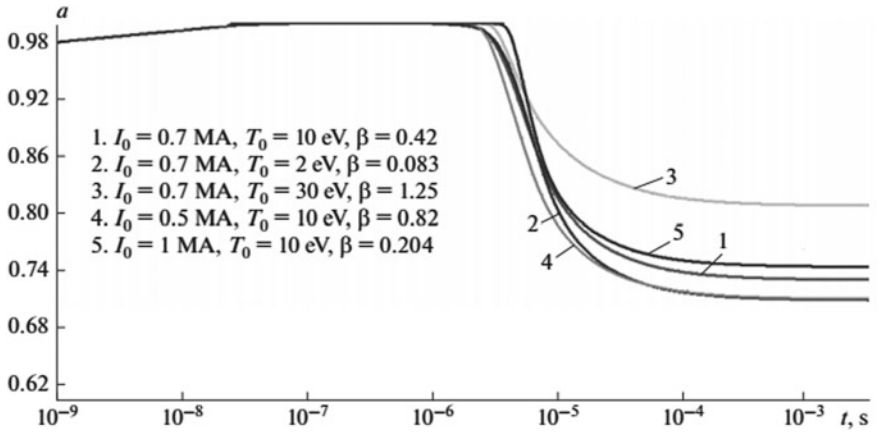


Fig. 7.5 Behavior of α at different initial conditions

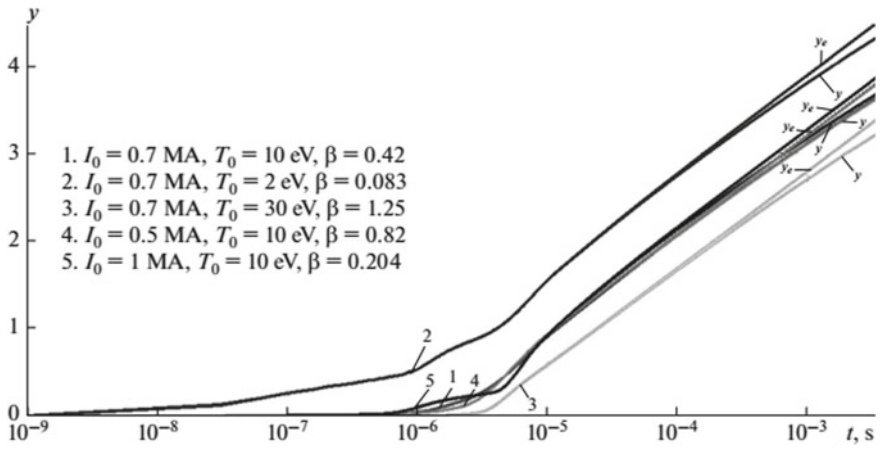


Fig. 7.6 Behavior of y and y_e at different initial conditions

currents $I_{0\mathrm{cr}} = 0.5 - 1\,\mathrm{MA}$. The decrease in $T_{0\mathrm{cr}}$ as compared to the adiabatic case is related to the release of the ionization energy.

7.6 Ionization Effect During the Propagation of TPB in the Ionosphere

Based on detailed numerical studies of the initial stage of the formation and movement of TPB after departure from the plasma gun, the problem of its further movement in the ionosphere and the assessment of the ionization effect that it exerts on the ionosphere were of practical interest.

There are presently not enough general expressions describing the energy loss by particles, both in the low and high energy regions. However, theoretical analysis provides the main qualitative conclusions about the dependence of energy loss $\mathrm{d}E/\mathrm{d}x$ on the velocity of particles [7]. The detailed analysis of theoretical models for estimating the rate of energy loss by a particle $\mathrm{d}E/\mathrm{d}x$ in the range of motion velocities s was given in [6]. At very low energies of incident particles (several tens of eV), the energy losses for protons are mainly described by elastic collisions with target atoms.

If the energy of the incident particle exceeds

$$E > 0.525\left[\left(z_{1a}^{2/3} + z_{2a}^{2/3}\right)\frac{M_1}{M_1 + M_2}\right]^2 \approx 0.04\ \mathrm{keV} = 40\ \mathrm{eV},$$

where $\frac{M_1}{M_1+M_2} = \frac{1}{1+16} = 0.058$, $z_{1a}^{2/3} + z_{2a}^{2/3} = 1 + 7.2^{2/3} = 4.75$, then the energy losses are mainly determined by inelastic collisions [7].

We used the results of the classical works of Firsov and Beta Bloch, as well as, experimental data for the energy loss function $F(u) = \frac{1}{n_a}\left|\frac{dE}{dx}\right|$. To pass protons through atomic oxygen [7], an approximation expression was obtained for $F(u)$ [6]. It is in satisfactory agreement with experimental data. Based on it, the change in the velocities (Fig. 7.7) and the concentration of electrons formed after passing through TPB was calculated (Fig. 7.8):

$$n_e(x) = \frac{N_p}{\pi^2 R(x) a(x)} \frac{F}{W_e},$$

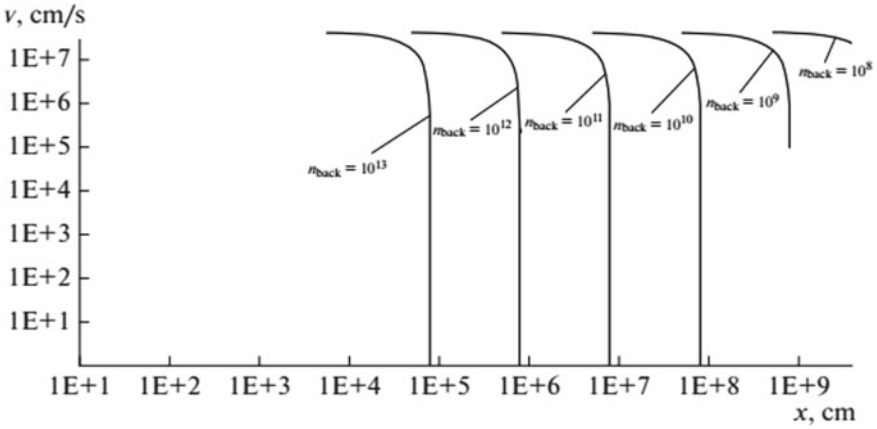


Fig. 7.7 Change of the particle velocity $v(x, n_f)$

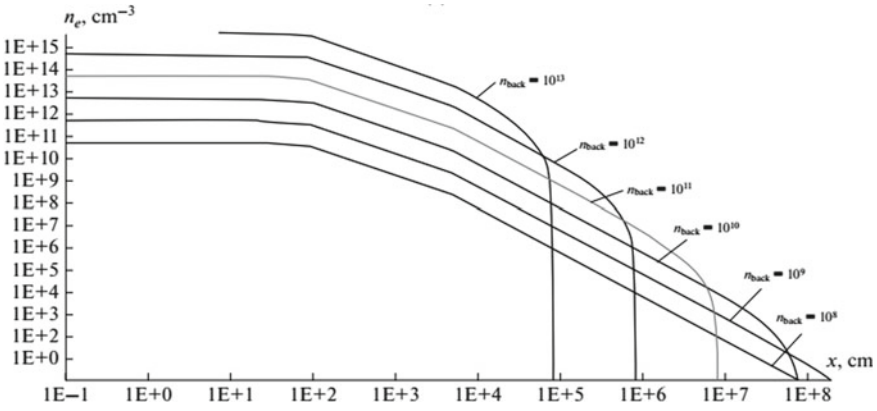


Fig. 7.8 Change of the electron concentration

where $N_p = 1.31 \times 10^{21}$ is the number of protons in TPB, $W_e = 33$ eV is the average energy required to form one electron–ion pair.

The deceleration of particles is sharp nature at all background concentrations. The electron concentration in the region of TPB passage significantly exceeds the background. The effect increases sharply with increasing proton velocity both in range and in electron concentration at the corresponding height.

The work showed that further improvement of the hydrogen-based plasma gun will achieve a plasma flow rate of 5×10^8 cm/s or more. This is important for the application of plasma guns in the issue of the effect of a rarefied plasma flow on the barrier.

7.7 The Effect of a Rarefied Plasma Flow on the Barrier

The present state of research on the effects on materials of pulsed beams of high particle density is given in [8].

Many sensitive elements of spacecraft are protected from possible environmental influences by glasses, which at the same time impose certain requirements on their optical properties—the ability to transmit radiation in a certain spectral range. The aim of this work is to use the molecular dynamics method for the numerical study of the effect of a rarefied plasma flow of a certain duration on a crystalline and amorphous structure, such as glass, to assess the nature and size of emerging defects and their effect on the transmission of light. For the model of continuous deceleration as a result of inelastic collisions, the Lindhard–Sharf theory gives good agreement with experiment [7]. However, to determine the structure of dislocations arising in a solid, a more detailed approach to the problem of energy dissipation by ions is needed. In [9], a numerical algorithm was developed for calculating the spatial motion of particles in a crystalline and amorphous body based on the presented molecular dynamics. A hybrid model was used for the collisional interaction of particles: if the limiting parameter $\rho \leq R_1 + R_2$, then the model of elastic balls was used. If $\rho > R_1 + R_2$, then the Coulomb interaction was assumed (R_1, R_2 is the radius of the particles). It is important that the multiplication of particles involved in the destruction of the crystal lattice is taken into account.

Figure 7.9 presents the effect of destruction of the Si crystal lattice at dissimilar energies of the hydrogen atom and its trajectory of motion.

Numerous calculations were performed for various types of plasma particles, their various energies and various amorphous and crystalline elements of a solid. The algorithm also takes into account the possibility of several falling particles entering the same cavity. Based on the results obtained, light scattering ($\lambda = 0.5$ μm) was studied on a defective surface of a transparent body. It is shown that as a result of the impact of TPB on protective glasses, they scatter from 20 to 50% of the incident light flux, which makes their functional purpose impossible as a means of protecting guiding devices.

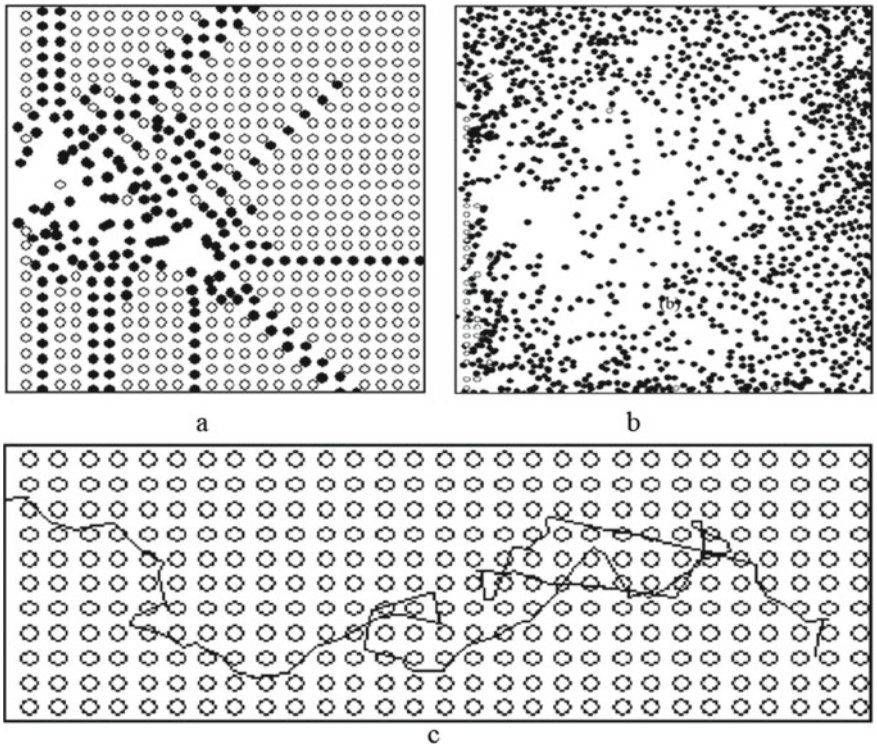


Fig. 7.9 Effect of destruction of the Si crystal lattice: **a** the resulting defect when a particle of energy $\varepsilon_0 = 10$ keV is hit, **b** the resulting defect when a particle of energy $\varepsilon_0 = 1000$ keV is hit, **c** trajectory of a plasma particle of energy $\varepsilon_0 = 1000$ keV

Thus, a complex study of both scientific and practical issues of the creation and study of plasma guns was carried out in the work.

7.8 Conclusions

This work presents a sufficiently detailed numerical study of TPB parameters at the initial stage of TPB motion and during further flight and interaction with strongly rarefied air. The motion distance depends on the air density and can reach ten kilometers in the upper ionosphere.

In this work, it was also shown that plasma guns actively developed at present can serve as an effective tool for the destruction of glass coatings. Further theoretical and experimental work in this area will make it possible to choose the optimal parameters for both the plasma device and the plasma itself. The performed studies, in particular, show that, in addition to a high velocity of the plasma flow, it is necessary to achieve

the minimal ionization degree that will provide the minimal decelerating action of the geomagnetic field on its dynamics.

Preliminary calculations also demonstrate that the degree of ionosphere ionization can significantly increase in the region of TPB propagation. At present, such studies are carried out in a sufficiently detailed formulation.

References

1. Degnan, J.H., Peterkin, R.E., Jr., Baca, G.P., Beason, J.D., Bell, D.E., Dearborn, M.E., Dietz, D., Douglas, M.R., Englert, S.E., Englert, T.J., Hackett, K.E., Holmes, J.H., Hussey, T.W., Kiuttu, G.F., Lehr, F.M., Marklin, G.J., Mullins, B.W., Price, D.W., Roderick, N.F., Ruden, E.L., Sovinec, C.R., Turchi, P.J.: Compact toroid formation, compression, and acceleration. Phys. Fluids **5**, 2938–2959 (1993)
2. Repin, A.Yu., Stupitskii, E.L., Shapranov, A.V.: Dynamics of a toroidal plasma cluster and its interaction with an obstacle. Ionization and dynamic characteristics and electromagnetic radiation. High Temp. **42**(4), 523–538 (2004)
3. Brode, H.L.: Review of nuclear weapons effects. Annu. Rev. Nucl. Sci. **18**, 153–202 (1968)
4. Bannov, S.G., Zhitlukhin, A.M., Cherkovets, V.E., Stupitsky, E.L., Motorin, A.A., Kholodov, A.S.: Dynamics of the plasma bunch at the initial and following stages of motion in a rarefied gas. Geomagn. Aeron. **59**(3), 318–341 (2019)
5. Stupitskii, E.L., Lyubchenko, O.S., Khudaverdyan, A.M.: Nonequilibrium processes accompanying expansion of a high-temperature plasma bunch. Sov. J. Quantum Electron. **15**(5), 682–688 (1985)
6. Moiseeva, D.S., Motorin, A.A., Stupitsky, E.L.: Assessment of the ionization effect during the distribution of a toroidal plasma bunch in a diluted atmosphere. Geomag. Aeron. **59**, 448–457 (2019)
7. Gott, Yu.V.: Interaction of particles with matter in plasma researches. USSR: Web (1978)
8. Boiko, V.I., Skvortsov, V.A., Fortov, V.E.: Interaction of pulsed beams of charged particles with matter. Fizmatlit, Moscow (2003) (in Russian)
9. Smirnov, E.V., Stupitskii, E.L.: Numerical simulation of the effect of rarefied plasma flow on the solid surface. J. Surf. Invest.: X-ray Synchrotron Neutron Tech. **4**(6), 965–975 (2010)

Chapter 8
Some Aspects on Pulsating Detonation Wave Numerical Simulation Using Detailed Chemical Kinetics Mechanism

Alexander I. Lopato

Abstract The chapter is dedicated to the numerical study of pulsating gaseous detonation wave propagation. The mathematical model is based on the Euler equations written for the multicomponent gas and supplemented by the detailed chemical reactions model to describe the combustion of the hydrogen–air mixture. The Petersen and Hanson kinetics is applied as the detailed chemical model. The numerical algorithm is based on the finite volume approach, essentially non-oscillatory scheme, AUSM numerical flux and the Runge–Kutta method. The numerical investigation of pulsating detonation wave propagation with direct detonation initiation near the closed end of the channel is carried out. The peculiarities of high-frequency and high-amplitude pulsations modes are discussed.

8.1 Introduction

Detonation wave (DW) is a supersonic complex consisting of a leading shock wave (LSW) followed by a chemical reaction zone. Detonation is a hydrodynamic wave process of propagation of an exothermic reaction through a substance at supersonic speed. Among the works on the study of detonation processes in gases, there are several directions. One of the directions includes the works on studies of detonation propagation in terms of safety engineering in tunnels and mines, where explosions and propagation of detonation and combustion waves are possible. Another direction involves the works on the study of detonation initiation and interaction of DWs in channels, stars, and other objects from the scientific point of view. As a third direction, note the works on detonation application in industry, including pulse-detonation gas burners and engines of the next generation, such as pulse detonation engines [1]. The development of the direction can be explained by the fact that detonation combustion is a thermodynamically advantageous method of fuel combustion and conversion of

A. I. Lopato (✉)
Institute for Computer Aided Design of the RAS, 19/18 Vtoraya Brestskaya ul., Moscow 123056, Russian Federation
e-mail: lopato2008@mail.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_8

chemical energy of the fuel into useful work. Thus, one can note the relevance of studies on detonation processes in gases.

Conducting natural experiments with the study of DWs is often associated with certain problems. The problems include the measurement of flow parameters and the stability of installations and sensors to high pressures and temperatures in areas with shock and detonation waves. In addition, the range of investigated properties of DWs is limited by a set of acceptable installations and constructions that can be used for conducting natural experiments. Numerical calculations are devoid of such problems and make it possible to obtain flow patterns with DWs in a sufficiently wide set of research areas with a degree of accuracy determined by a number of factors including the adequacy of the mathematical model and the numerical method, the approximation order of the numerical scheme, and the stability of the numerical method.

The chapter is organized as follows. Related work is highlighted in Sect. 8.2. Section 8.3 provides the mathematical model of the studied problem. The computational algorithm of the second approximation order is described in Sect. 8.4. Section 8.5 presents the results of verification and numerical experiments. Section 8.6 concludes the chapter.

8.2 Related Work

Since DW, in general, is a multi-dimensional object, mathematical modeling requires taking into account multidimensional effects with a complex kinetics model of chemical reactions. On the other hand, the mathematical model corresponding to the one-dimensional structure of DW is simpler but provides a relatively rich spectrum of dynamic features that deserve the detailed study and have relevance to multidimensional effects of DW, for example, cellular structures. As is known from numerical and experimental studies, the propagation of DW is associated with the formation of a complex nonlinear oscillating process including pulsations of parameters behind the front of DW, which are investigated in a number of numerical studies. In [2, 3], different modes of parameters pulsations of the one-dimensional DW were obtained depending on the activation energy of the considered model mixture. The mathematical model in [4] included the one-step irreversible reaction and Arrhenius kinetics with parameters values that were proposed apparently for the first time. A spectral Fourier analysis of the peak pressure pulsations on time was carried out with the allocation of dominant frequencies. The comparison with the results of theoretical and numerical studies on some quantitative characteristics such as limit cycle size in the case of the weakly unstable detonation was performed.

In [5, 6], detonation in the model hydrogen–air mixture [7] was considered. High-frequency (HF) and low-frequency modes of detonation propagation were obtained in [5]. It is shown that with an increase in the approximation order of the numerical method, the front of the reaction zone is less smeared and the scale of the reaction zone is better resolved, which leads to that the instabilities are captured correctly.

This tendency was confirmed with the results from [2]. In [6], the mechanism accompanying the process of detonation propagation in the channel with an array of circular obstacles was studied. The mechanism was based on the formation of a temperature gradient and spontaneous waves in the unburned gas adjacent to DW front. The influence of chemical kinetics on the modeling of detonation initiation by a temperature gradient in the hydrogen–air mixture was discussed in [8]. The chemical model that contains 19 reactions and 9 components was considered as the detailed kinetics model. The model described correctly some parameters like ignition delay times and laminar flames characteristics for a wide range of initial parameters. The global Arrhenius kinetics [7] of hydrogen–air combustion was considered as the one-step model. The model reproduced some characteristics like the flame speed and the width of the laminar flame. The critical size of the "hot spots" capable to initiate detonation was shown to be larger in the case of the detailed chemistry. The differences were explained by the fact that the one-step kinetics model is exothermic for all temperatures, while chain branching reactions in complex kinetics start with endothermic induction stage representing chain initiation and branching. Besides, the induction times obtained using one-step kinetics were several orders of magnitude smaller than the experimental results that are in good agreement with induction times obtained using detailed kinetics. Thus, the complex kinetics is shown to provide a sufficiently wide range of parameters, where the kinetics works correctly, and takes into account a number of factors better than the one-step one, especially at the stage of detonation initiation, although it significantly increases the calculation time. On the other hand, the use of one-step kinetics with gasdynamics values from a vicinity of parameters that are used in the process of calibrating gives an opportunity to obtain some adequate results and useful recommendations to study the dynamics of detonation instability.

In [9], the detailed analysis of the nonlinear dynamics of detonation in the hydrogen–air mixture was carried out using the detailed kinetics model. The mathematical model included the system of Euler equations written for the case of the multicomponent mixture. The chemical kinetics model included nine components and 38 elementary reactions. The numerical method of the high order of accuracy included the fifth-order convergence rate monotonicity preserving scheme, the third-order total variation diminishing the Runge–Kutta time integration scheme, Roe flux and Gaussian elimination scheme for solving chemical kinetics implicitly. The authors considered direct initiation of 1D detonation in the channel covered with computational grids with cell sizes of 2.5 and 12.5 μm. The transition from the overdriven detonation regime to the self-sustaining one with the formation of two pulsating modes was obtained. For both grid sizes, the HF pulsations mode was followed by the high-amplitude (HA) pulsations mode with the time increase. The specific features of each mode including the frequency values are described. The mechanism of processes in the induction zone behind the front of LSW was described in terms of acoustic and entropy waves in a manner similar to that of McVey and Toong [10] and it seems to be a reasonable description of the mechanism of pulsating detonation. The work demonstrated a sensitivity of the results to the values of the initial conditions parameters, grid resolution, and properties of the numerical method.

The aim of current work is the mathematical modeling of pulsating DW propagation in the hydrogen–air mixture using the numerical method of the second approximation order and detailed chemical kinetics model.

8.3 Mathematical Model

Mathematical model is based on the one-dimensional system of Euler equations written in the laboratory frame for the case of multicomponent media and supplemented by the detailed chemical kinetics model:

$$\begin{gathered}
\tfrac{\partial \mathbf{U}}{\partial t} + \tfrac{\partial \mathbf{F}}{\partial x} = \mathbf{S}, \\
\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ e \\ \rho Y_s \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ (e+p)u \\ \rho Y_s u \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \rho \omega_s \end{bmatrix}, \\
e = \sum_{s=1}^{\mathrm{NS}} \tfrac{\rho Y_s}{\mu_s} h_s(T) - p + \tfrac{\rho u^2}{2}, \quad p = \sum_{s=1}^{\mathrm{NS}} \tfrac{\rho Y_s}{\mu_s} RT, \\
\omega_s = \mu_s \sum_{j=1}^{\mathrm{NR}} \left(\sum_{l=1}^{\mathrm{NS}} \alpha_{lj} c_l \right)^{\beta_j} \left(\gamma''_{js} - \gamma'_{js} \right) \left[K_{fj} \prod_{i=1}^{\mathrm{NS}} c_i^{\gamma'_{js}} - K_{bj} \prod_{i=1}^{\mathrm{NS}} c_i^{\gamma''_{js}} \right], \quad c_i = \tfrac{\rho Y_i}{\mu_i}.
\end{gathered} \tag{8.1}$$

Here, ρ is the total mixture density, u is the velocity, p is the pressure, e is the total energy density, R is the universal gas constant, Y_s is the mass fraction of the mixture component s, ω_s is the production rate, h_s is the molar enthalpy, μ_s is the molecular weight, c_i is the molar concentration, α_{lj} is a third body coefficient, γ'_{js} and γ''_{js} are the stoichiometric coefficients, K_{fj} is the forward rate constant, K_{bj} is the backward rate constant, NS is the total number of components, NR is the total number of reactions. The molar enthalpy is calculated as

$$h_s(T) = RT\left(a_{1s} + \frac{a_{2s}}{2}T + \frac{a_{3s}}{3}T^2 + \frac{a_{4s}}{4}T^3 + \frac{a_{5s}}{5}T^4 + \frac{a_{6s}}{T}\right),$$

where the coefficients $a_{1s}, \ldots, a_{6s}$ are presented in [11]. The specific heat ratio of the multicomponent mixture in such mathematical model depends on the temperature as

$$\gamma(T) = 1 + R\frac{\sum_{s=1}^{\mathrm{NS}} Y_s / \mu_s}{\sum_{s=1}^{\mathrm{NS}} Y_s \left(C_{ps}(T) - R\right) / \mu_s},$$

where C_{ps} is the molar heat capacity at constant pressure of the components

$$C_{ps}(T) = R\left(a_{1s} + a_{2s}T + a_{3s}T^2 + a_{4s}T^3 + a_{5s}T^4\right).$$

The sound velocity is calculated for the multicomponent mixture using the formula:

$$c = \sqrt{RT \frac{\sum_{s=1}^{NS} \frac{Y_s C_{ps}(T)}{\mu_s} \sum_{s=1}^{NS} \frac{Y_s}{\mu_s}}{\sum_{s=1}^{NS} \frac{Y_s}{\mu_s} \left(C_{ps}(T) - R\right)}}.$$

In current work, the Petersen and Hanson (PH) model [12] is used to describe the combustion and detonation in the stoichiometric hydrogen–oxygen mixture. The model takes into account NS = 9 components (H_2, O_2, H, O, OH, HO_2, H_2O_2, H_2O, and N_2) and NR = 18 elementary reactions. The stoichiometric coefficients γ'_{js} and γ''_{js}, rate constants K_{fj} and K_{bj}, third body coefficients α_{lj} can be found in [12]. The applicability and efficiency of kinetics for numerical simulations of chemical reactions in hydrogen–air and hydrogen–oxygen mixtures is confirmed by a number of works (see, for example, references in [13]).

8.4 Numerical Algorithm

The computational algorithm is based on the Strang splitting principle in terms of physical processes [14]. When passing from one time layer to another one, one first integrates the gas dynamics equations without considering the chemical reactions ($\mathbf{S} = 0$, see Eq. 8.1), and, thereby, performs the first stage of the splitting procedure. Then, one estimates the contribution of the chemical reactions without considering the convection (the second stage of splitting).

The spatial part of Eq. 8.1 is discretized using the finite-volume method

$$\frac{\partial \mathbf{U}}{\partial t} = -\frac{\mathbf{F}_{i+1/2} - \mathbf{F}_{i-1/2}}{\Delta x} = \mathbf{L}_i(\mathbf{Q}).$$

Here, i is the index of computational grid cell. Indexes $i + 1/2$ and $i - 1/2$ denote right and left bounds of cell i, respectively, Δx is a cell size, $\mathbf{Q}$ is the unknown grid function, $\mathbf{F}$ is a numerical flux. The numerical flux $\mathbf{F}_{i+1/2}$ is calculated using AUSM numerical scheme [15] extended to the case of a multicomponent gas mixture:

$$\mathbf{F}_{i+1/2} = \frac{1}{2} M_{1/2} \left[\mathbf{W}\left(\mathbf{Q}_i^+\right) + \mathbf{W}\left(\mathbf{Q}_{i+1}^-\right)\right] + \frac{1}{2} |M_{1/2}| \left[\mathbf{W}\left(\mathbf{Q}_i^+\right) - \mathbf{W}\left(\mathbf{Q}_{i+1}^-\right)\right] + \mathbf{P}_{1/2},$$

where

$$M_{1/2} = M_i^+ + M_{i+1}^-, \ \mathbf{W}(\mathbf{Q}_i) = \begin{pmatrix} \rho c \\ \rho u c \\ \rho H c \\ \rho Y_s c \end{pmatrix}_i, \quad \mathbf{P}_{1/2} = \begin{pmatrix} 0 \\ p_{1/2} \\ 0 \\ 0 \end{pmatrix}, \quad p_{1/2} = p_i^+ + p_{i+1}^-.$$

Here, $H = e + p$ is the enthalpy of a volume unit of the mixture. The upper index "+" corresponds to the parameters on the right bound of the cell i, while the index "−" corresponds to the left bound of the cell $i + 1$. The elements of $\mathbf{Q}_i^+$, $\mathbf{Q}_{i+1}^-$ are determined using the ENO-reconstruction of the second approximation order of the conservative variables [16]. The parameters M_i^+, M_{i+1}^-, p_i^+, p_{i+1}^- are calculated in accordance with [15]. Note that the use of the AUSM scheme in the calculations is not an obligatory requirement. Thus, in [17, 18], numerical modeling of combustion and detonation waves was carried out using the Courant-Isaacson-Rees flux scheme.

For the temporal discretization, the second-order Runge–Kutta scheme is applied [19]:

$$\begin{cases} \mathbf{Q}_i^{(1)} = \mathbf{Q}_i^n + \Delta t^n \cdot \mathbf{L}_i(\mathbf{Q}^n), \\ \tilde{\mathbf{Q}}_i^{n+1} = \frac{1}{2}\mathbf{Q}_i^n + \frac{1}{2}\mathbf{Q}_i^{(1)} + \frac{1}{2}\Delta t^n \cdot \mathbf{L}_i\left(\mathbf{Q}^{(1)}\right), \end{cases}$$

where Δt^n is a time step that is chosen dynamically from the stability condition. The upper tilde indicates that the solution obtained in this way is the result of the first stage of the splitting procedure of physical processes.

On the second stage, the chemical reactions are taken into account without considering the convection (the second stage of splitting). The stage involves the solving of the system of ordinary differential equations which describes the chemical reactions kinetics for the molar concentrations and temperature in each computational grid cell:

$$\frac{\mathrm{d}c_s}{\mathrm{d}t} = \sum_{j=1}^{\mathrm{NR}} \left(\sum_{l=1}^{\mathrm{NS}} \alpha_{lj} c_l \right)^{\beta_j} \left(\gamma''_{js} - \gamma'_{js} \right) \left[K_{fj} \prod_{i=1}^{\mathrm{NS}} c_i^{\gamma'_{js}} - K_{bj} \prod_{i=1}^{\mathrm{NS}} c_i^{\gamma''_{js}} \right], \quad s = 1, 2, \ldots, \mathrm{NS},$$

$$\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{RT \sum_{s=1}^{\mathrm{NS}} \frac{\mathrm{d}c_s}{\mathrm{d}t} - \sum_{s=1}^{\mathrm{NS}} \left(h_s(T) \frac{\mathrm{d}c_s}{\mathrm{d}t} \right)}{\sum_{s=1}^{\mathrm{NS}} \left(c_s \left(C_{ps}(T) - R \right) \right)}.$$

The system is integrated on the time step Δt^n. According to the splitting method, initial conditions of the system are taken from the solution of the first gas-dynamic stage. The system is solved with the use of the implicit Euler method with Newton linearization.

The computational algorithm, noted above, is based on the algorithm constructed for the case of the two component mixture (reagent and product), and constant value of the specific heat ratio [20]. In this work, the procedure was extended for the case of the multicomponent mixture with noted dependence $\gamma(T)$.

8.5 Verification and Results

The estimation of the practical approximation order of the algorithm in the case of the two-component mixture and constant value of the specific heat ratio in the work

[3] gives the value near 2. To verify the realized PH chemical reaction model, the 0D homogeneous ignition simulation for the stoichiometric hydrogen–air mixture was carried out. Figure 8.1 demonstrates the time dependence of the mass fractions of the components at the initial pressure 1 atm and temperature 1000 K. The obtained dependencies are similar with the results from [13]. Relatively small quantitative differences (the calculated ignition time is equal to 210 μs, while the value in [13] is about 225 μs) can be explained by the fact that the polynomial coefficients $a_{1s}, a_{2s}, \ldots$ in [13] that used in calculations of the heat capacities and other thermodynamic properties were taken from another database. We did not try to achieve a complete agreement in the results. It was important to establish that the chemical kinetics model is realized correctly and can be used for modeling of complex problems, including combustion and detonation in hydrogen–air and hydrogen–oxygen mixtures.

Let us consider the numerical results of propagation of the pulsating DW with the use of the mathematical model and computational algorithm noted above. Direct detonation initiation in the work is simulated in the channel of length $L = 3$ m filled with a resting stoichiometric hydrogen–air mixture. Detonation is initiated as a result of instantaneous energy release in a short region of the length $l = 10$ cm adjacent to the left boundary of the channel and named the spark region. In this region, the high pressure $p_l = 10$ atm and temperature $T_l = 3000$ K are set at the initial time moment. The rest of the channel is filled with the mixture under the pressure $p_0 = 0.1$ atm and temperature $T_0 = 300$ K. The area of the channel is covered by a computational grid with the cell size $\Delta x = 25$ μm.

Figure 8.2 depicts the dynamics of the variation of the peak pressure in the calculated region and shows the process of initiation and propagation of DW. At the initial stage of the computation, the mixture in the spark region is burned instantaneously

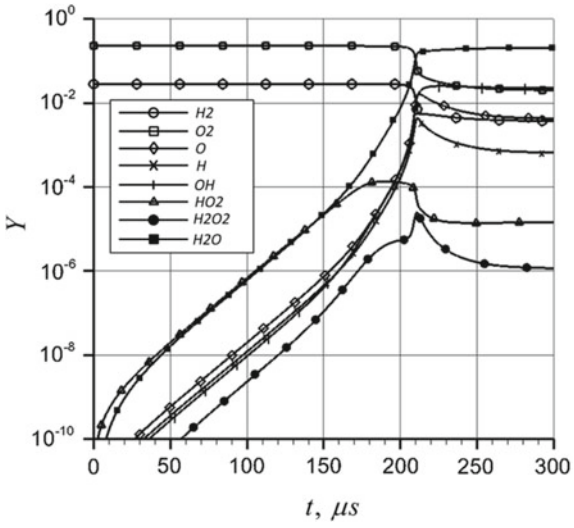


Fig. 8.1 Dependences of mass fractions of the mixture components on time for initial pressure $p = 1$ atm and temperature $T = 1000$ K

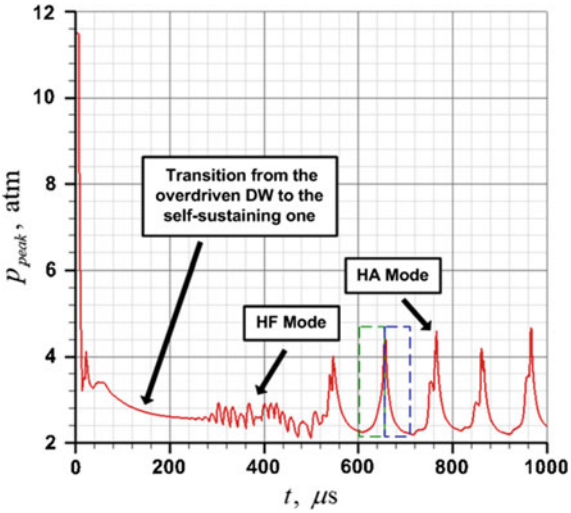


Fig. 8.2 Predicted peak pressure time history in the computational domain

and increases gasdynamics parameters. The mixture with high pressure and temperature generates the combustion wave that propagates to the right and interacts with the leading wave with its amplification. The formation of the overdriven DW takes place at the time moment of about 50 μs that can be considered as the end of the initial stage. After the initial stage, DW obtains the classical structure with LSW, reaction zone and Taylor expansion wave. DW remains overdriven and then the transition from the overdriven to self-sustaining regime occurs. During the transition, the velocity of DW exceeds the value of 1900 m/s. Note that the theoretical value of the Chapman-Jouguet (CJ) velocity specified in [7] is equal to 1993 m/s. So, the values of the CJ velocity are in good agreement with each other. As the detonation approaches the CJ state (the time moment of about 240 μs), pulsations of the HF mode begin to appear, as shown in Fig. 8.2. The pulsations of parameters are associated with the interaction between the combustion waves formed in the reaction zone and LSW. The HF mode is characterized by a relatively small distance and change in distance from DW front to the reaction zone. Further development of the pulsations leads to the transition from the HF mode to the HA one at the time of about 520 μs. As shown in Fig. 8.2, the signal of the pulsations in the HA mode is close to periodic. Figure 8.3 shows the time evolution of temperature and density profiles every 10 μs during the time interval 620–660 μs. The acceleration of the chemical rates in the reaction zone enhances the rate of heat release. The formation of the combustion wave occurs. The accelerating flame burns the mixture in the induction zone releasing large amount of energy up to the collision of the combustion wave with LSW. Thus, at this stage, the formed disturbance propagates toward DW front, and the reduction of the induction zone occurs. The stage corresponds to the region marked with a green dashed line in Fig. 8.2 and is referred to as an acoustic wave cycle. The next considered stage corresponds to the time interval 660–700 μs. The stage is marked with a blue dashed line in Fig. 8.2 and is called an entropy wave

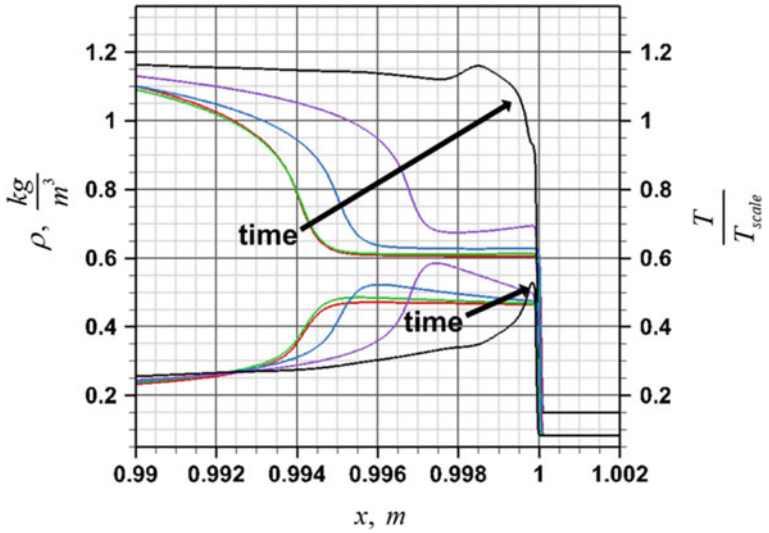


Fig. 8.3 Temperature (above) and density (below) distributions in shock reference frame at the successive time moments within the acoustic wave cycle. The time interval is 620–660 μs. The arrows show the order of profiles in time. $T_{\text{scale}} = 2000$ K

cycle. Figure 8.4 shows the time evolution of temperature and density profiles every 10 μs during the time interval 660–700 μs. At this stage the disturbance propagates back into the induction zone and the attenuation of LSW occurs as shown in Fig. 8.4. The cycle is also characterized by an increase in the size of the induction zone and the formation of conditions for a next acoustic cycle. Note that the obtained density and temperature profiles qualitatively well correlate with the results from [9].

One can also see that the HA mode includes the period-doubling effect, i.e., the dual oscillations in the peak pressure history are observed. For example, the time interval 700–800 μs contains two local maximum values of pressure. The first one is associated with the time moment of about 750 μs and the pressure value 3.3 atm, and corresponds to the mechanism discussed above. Consider the time interval 760–767 μs that is associated with the second maximum. Figure 8.5 demonstrates the distributions of pressure and mass fraction of the H_2 component every 1 μs in this time interval. After the collision of the combustion wave with LSW the reaction of combustion of the mixture occurs directly behind LSW with the formation of a pocket of partly burnt gas. The burning of the mixture in the pocket at some distance behind LSW leads to a new combustion wave and an increase in temperature and pressure of the gas. As a result, the second local maximum of pressure corresponds to the time moment of about 766 μs in Fig. 8.2 and has the value of about 4.6 atm. The dual oscillations of the HA mode in other time intervals in Fig. 8.2 takes place in accordance with the mechanism noted above.

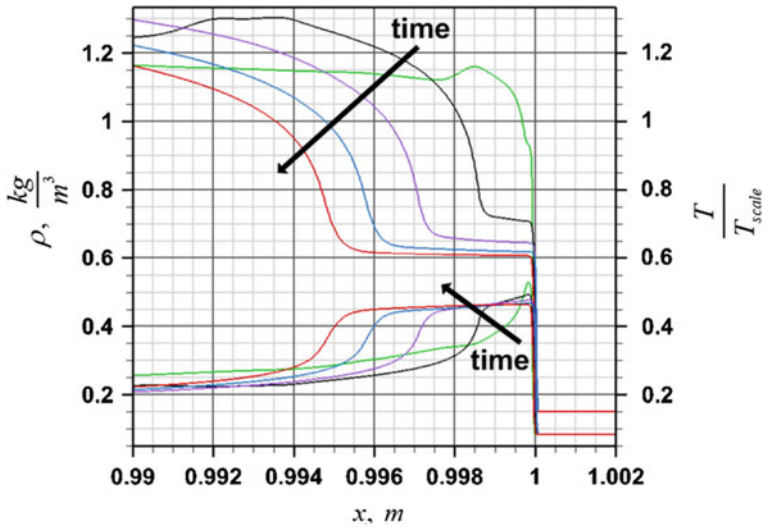


Fig. 8.4 Temperature (above) and density (below) distributions in shock reference frame at the successive time moments within the entropy wave cycle. The time interval is 660–700 μs. The arrows show the order of profiles in time $T_{\text{scale}} = 2000\,\text{K}$

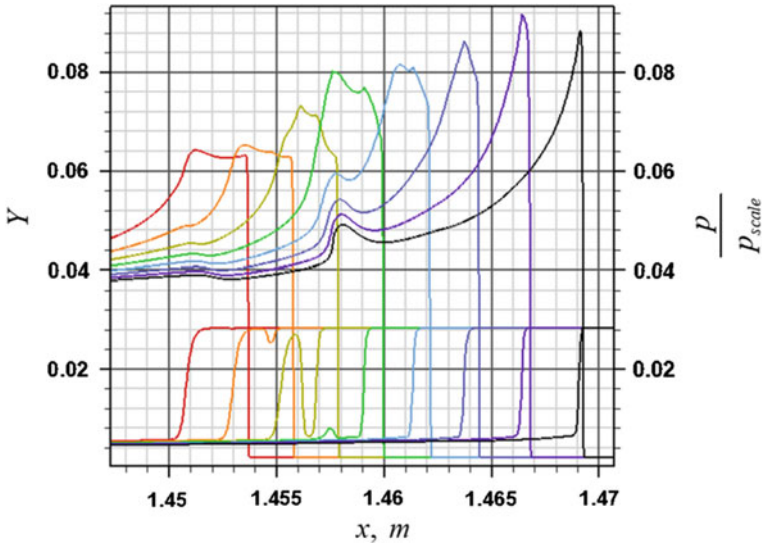


Fig. 8.5 Distributions of pressure (above) and mass fraction of the component H_2 (below) at the successive time moments in the time interval 760–767 μs. $p_{\text{scale}} = 50\,\text{atm}$

8.6 Concluding Remarks

The work demonstrates the possibility of numerical modeling of initiation and propagation of gaseous detonation waves using complex reaction kinetics. The mathematical model is based on the system of Euler equations for a multicomponent media. The computational algorithm of the second approximation order based on the physical processes splitting technique, finite volume method, ENO-reconstruction, and AUSM flux are described. To verify the implemented computational algorithm, the 0D ignition simulation was performed.

The numerical investigation of pulsating detonation wave propagation with the use of Petersen and Hanson detailed kinetics corresponding to the stoichiometric hydrogen–air mixture was carried out. The direct detonation initiation near the closed end of the channel was considered. The cell size of the computational grid was sufficient to obtain the pulsations of various modes. During the computation, the high-frequency and high-amplitude pulsation modes were obtained. The characteristics and frequencies are both very different. The border separating the modes is clearly defined. The features of both modes, as well as, the processes accompanying pulsations are considered. The induction zone in the high-frequency mode fluctuates without significant change in contrast to the high-amplitude. The signal of the pulsations in the high-amplitude mode is close to periodic.

Acknowledgements This work is carried out under the state task of the ICAD RAS.

References

1. Kailasanath, K.: Review of propulsion applications of detonation waves. AIAA J. **38**(9), 1698–1708 (2000)
2. Sharpe, G.J., Falle, S.A.: Numerical simulations of pulsating detonations: I. Nonlinear stability of steady detonations. Combust. Theory Model. **4**, 557–574 (2000)
3. Lopato, A.I., Utkin, P.S.: Toward second-order algorithm for the pulsating detonation wave modeling in the shock-attached frame. Combust. Sci. Technol. **188**, 1844–1856 (2016)
4. Lee, H.I., Stewart, D.S.: Calculation of linear detonation instability: one-dimensional instability of plane detonation. J. Fluid Mech. **216**, 103–132 (1990)
5. Lopato, A.I., Utkin, P.S.: Mathematical modeling of pulsating detonation wave propagation using monotone numerical methods of different approximation orders. In: Transient Combustion and Detonation Phenomena: Fundamentals and Applications, pp. 261–268. Torus Press, Moscow (2014)
6. Xiao, H., Oran, E.S.: Shock focusing and detonation initiation at a flame front. Combust. Flame **203**, 397–406 (2009)
7. Gamezo, V., Ogawa, T., Oran, E.: Flame acceleration and DDT in channels with obstacles: effect of obstacle spacing. Combust. Flame **155**, 302–315 (2008)
8. Liberman, M., Wang, C., Qian, C., Liu, J.: Influence of chemical kinetics on spontaneous waves and detonation initiation in highly reactive and low reactive mixtures. Combust. Theory Model. **23**(3), 467–495 (2019)
9. Cole, L.K., Karagozian, A.R., Cambier, J.-L.: Stability of flame-shock coupling in detonation waves: 1D dynamics. Combust. Sci. Technol. **184**, 150–1525 (2012)

10. McVey, J.B., Toong, T.Y.: Mechanism of instabilities of exothermic hypersonic blunt-body flows. Combust. Sci. Technol. **3**, 63–76 (1971)
11. Burcat thermochemical data. https://burcat.technion.ac.il. Accessed 03 Jul 2020
12. Petersen, E.L., Hanson, R.K.: Reduced kinetics mechanisms for RAM accelerator combustion. J. Propuls. Power **15**(4), 591–600 (1999)
13. Togashi, F., Lohner, R., Tsuboi, N.: Numerical simulation of H_2/air detonation using unstructured mesh. Shock Waves **19**, 151–162 (2009)
14. Toro, E.F.: Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction, 3rd edn. Springer, Berlin (2009)
15. Liou, M.-S., Steffen Jr., C.J.: A new flux splitting scheme. J. Comput. Phys. **107**, 23–39 (1993)
16. Shu, C.-W.: Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. NASA/CR-97-206253, ICASE Report No. 97-65 (1997)
17. Lopato, A.I., Eremenko, A.G., Utkin, P.S., Gavrilov, D.A.: Numerical simulation of detonation initiation: the quest of grid resolution. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Theory and Practice of Computational Mechanics. SIST, vol. 173, pp. 79–89. Springer, Singapore (2020)
18. Lopato, A.I., Utkin, P.S: The usage of grid-characteristic method for the simulation of flows with detonation waves. In: Petrov, I.B., Favorskaya, A.V., Favorskaya, M.N., Simakov, S.S., Jain, L.C. (eds.) Smart Modeling for Engineering Systems. GCM50 2018. SIST, vol. 133, pp. 281–290. Springer, Cham (2018)
19. Shu, C.W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes. J. Comput. Phys. **77**, 439–471 (1988)
20. Lopato, A.I., Utkin, P.S: Numerical study of detonation wave propagation in the variable cross-section channel using unstructured computational grids. J. Combust. 3635797, 1–8 (2018)

Chapter 9
A Godunov-Type Method for a Multi-temperature Plasma with Strong Shock Waves and a General Equation of State

Alexey G. Aksenov

Abstract A multi-temperature code for a multi-component gas dynamic is considered. The velocities of components with nonzero mass are assumed to be identical to each other. The gas dynamic part is the Godunov-type method based on the efficient approximate solution of the Riemann problem operating with all components of the homogeneous gas mixture. The method assumes the table equation of state (EOS), but the system of the hydrodynamic equations should be hyperbolic. This work contains the test of the method on a strong shock wave in hydrogen plasma, so-called Shafranov's solution. By taking into account the radiation component, the chapter discusses the applicability of the two temperature models for the strong shock wave in the hydrogen with the large temperatures behind a shock wave without consideration of the radiation at a considered short timescale. General EOS for the mixture of protons, electrons, and radiation differs from an ideal gas low EOS for two components (protons and electrons) fully ionized hydrogen plasma.

9.1 Introduction

A multi-component gas of different substances α is described by a set of densities $\rho_\alpha(\mathbf{r}, t) \equiv c_\alpha(\mathbf{r}, t)\rho(\mathbf{r}, t)$, where c_α are concentrations, and internal energy densities $\rho\varepsilon_\alpha(\mathbf{r}, t)$, where ε_α is specific energy. All massive particles have identical velocities $\mathbf{v}(\mathbf{r}, t)$ and temperatures, while the "massless" fast particles from viewpoint of the total density ρ (electrons, photons) have their own temperatures. The equations of state are $P = \sum_\alpha P(\rho, \varepsilon_\alpha)$, $\varepsilon_\alpha = \varepsilon_\alpha(\rho, T_\alpha)$. The components can exchange energy, can transfer energy by heat conduction not associated with the transfer of the massive particles, and can participate in reactions. Such problems arise in inertial thermonuclear fusion [1], laser ablation experiments [2, 3], and astrophysics [4]. This is an intermediate case between the description based on the Boltzmann kinetic

A. G. Aksenov (✉)
Institute for Computer Aided Design of the RAS, 19/18, Vtoraya Brestskaya ul., Moscow 123056, Russian Federation
e-mail: aksenov@icad.org.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_9

equations for distribution functions $f_\alpha(\mathbf{r}, \mathbf{p}, t)$ and classical single-component gas dynamics. Taking into account possible large opacities for the fast particles, one should consider the gas for the mixture without transfer to the separate description of the gas dynamics of the matter and "massless" fast particles. The joined treatment allows to integrate the gas dynamic transport with the maximal possible time steps in the explicit scheme even at the optically thick cases.

Efficient Riemann problem solvers for such hydrodynamic equations are constructed, see, e.g., for some special case of EOS [5–7] and general EOS [8]. Below an original method based on the Riemann problem solver for the multi-temperature nonequilibrium gas [9–11] is briefly described. This method operates with homogenous mixture of components of the matter on the fixed Eulerian grid to carry out the deep phase of the development of the hydrodynamic instabilities. To improve the spatial resolution, it uses the reconstruction of the contact discontinuities on fixed grid as an original method [12]. The method was applied within the plasma physics for the inertial heavy-ion fusion [1, 13] and is useful in astrophysical tasks with hydrodynamic and the radiation transfer [14, 15]. In the local model for EOS proposed, it is assumed that the entropy variations in neighboring mesh cells are small at the evaluation of the dimensionless coefficients EOS from the pressure jump across the discontinuity. In the case of an arbitrarily large pressure jump, the model yields physically reasonable results. In real cases, the pressure jumps are not small on the surfaces of nearest cells of the computational grid without viscosity.

The simplest multi-temperature shock wave (SW) structure in plasma was considered in [16]. This solution is a suitable test for the method. This work prevents disadvantages of the strong SW test from the viewpoint of the physics in the last publications [9–11]. To provide the correct simple physical description, one starts from the heated ionized hydrogen plasma in the initial state. Thus, the temperature after the strong SW becomes huge enough for the important role of the disregarded radiation. Do the obtained mathematical results for strong SW contain physical applications? It is possible to give the answer on the base of qualitative estimates. Also by means of introducing nonequilibrium radiation into the developed code, it is possible to give the qualitative answer. EOS for the mixture of protons, electrons, and radiation is not ideal gas low in comparison with EOS for protons and electrons in fully ionized hydrogen plasma. The introducing of radiation also illustrates an application of the method to the general EOS. General EOS can contain domains with a negative square of the sound speed $c^2 \equiv (\mathrm{d}P/\mathrm{d}\rho)_s$ at phase transitions. In these domains, the gas enthalpy should be corrected to provide the nonnegative sound speed square.

The chapter is organized as follows. Section 9.2 provides a formulation of the problem and the numerical method. Shock wave structure in hydrogen plasma is discussed in Sect. 9.3. Discussion about radiation effects is given in Sect. 9.4. Section 9.5 concludes the chapter.

9.2 Formulation of the Problem and the Numerical Method

The system in the fixed Euler coordinates is: the mass transfer equations for the components

$$\frac{\partial \rho_\alpha}{\partial t} + \mathrm{div}\rho_\alpha \mathbf{v} = \rho \dot{c}_\alpha, \tag{9.1}$$

the momentum conservation law

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \mathrm{Div}\,\Pi = 0, \tag{9.2}$$

and the energy density equations

$$\frac{\partial \rho E_\alpha}{\partial t} + \mathrm{div}(\rho E_\alpha + P_\alpha)\mathbf{v} + \mathbf{v}(c_\alpha \mathrm{grad}\, P - \mathrm{grad}\, P_\alpha) = \mathrm{div}(\kappa \mathrm{grad}\, T_\alpha) + \rho Q_\alpha, \tag{9.3}$$

where the energy densities $E_\alpha = \varepsilon_\alpha + c_\alpha \mathbf{v}^2/2$, the tensor $\Pi_{ij} = \rho v_i v_j + P\delta_{ij}$, and the equation of state $P = \sum_\alpha P_\alpha(\rho, \mathbf{c}, \varepsilon_\alpha)$ with specific energies $\varepsilon_\alpha(\rho, \mathbf{c}, T_\alpha)$. The kinetic coefficients $\dot{c}_\alpha$, κ_α, and Q_α depend on ρ, $\mathbf{c}$, and $\mathbf{T}$. The problem is computed by applying the splitting on the physical processes and the dimensional splitting. The heat conduction equations are solved using central difference approximations. As a result, the system of partial differential equations (PDEs) is reduced to the ordinary differential equations (ODEs) system for $\dot{\varepsilon}_{\alpha,i}$. ODEs system is solved by applying the implicit Gear's method [17]. To describe the kinetics of reactions, ODEs system for $\dot{\rho}_\alpha$ and $\dot{\varepsilon}_\alpha$ is solved in each grid cell also by Gear's method. The transport of "massless" particles in both transparent and opaque cases can be described in the frame of diffusion with flux limiters.

The system in Euler variables involves the term $\mathbf{v}(c_\alpha \mathrm{grad}\, P - \mathrm{grad}\, P_\alpha)$, which is different from the divergence of a flux. This term requires a special treatment at discontinuities. In the classical problem of SW in a hydrogen plasma, one has only three conservation laws and, due to heat conduction, a piecewise-smooth temperature, for which a differential equation can be used instead of a conservation law.

The hydrodynamic part of the code is based on a high-order explicit Godunov scheme for single-temperature single-component gas dynamics [9, 12]. A local model for EOS simplifies the solution of the Riemann problem and makes it possible to obtain fluxes and partial pressures of the components in any flow region with discontinuities. Following [18], such model for a multi-component gas is constructed so as it holds strictly in the case of weak discontinuities. The increment of the specific entropy s across SW is a quantity of the third order of smallness with respect to the pressure jump: $O([P]^3)$. Neglecting the entropy variation behind SW, one computes the dimensionless coefficients $\gamma_\alpha \equiv P_\alpha \tau/\varepsilon_\alpha + 1$ as functions of the state ahead of SW and the total pressure behind SW. The local model for EOS is used

to solve the Riemann problem. On the left and right of the contact discontinuity, the concentrations remain constant, and EOS is independent of c_α. Using the relation $\mathrm{d}\varepsilon_\alpha(s_\alpha, \tau) = T_\alpha \mathrm{d}s_\alpha - P_\alpha \mathrm{d}\tau$ and the assumption $\mathrm{d}s_\alpha = 0$, one obtains explicit expressions for the non-dimensional adiabatic indexes (see [9])

$$\frac{\mathrm{d}\gamma_\alpha}{\mathrm{d}P_\alpha} = \frac{\tau}{\varepsilon_\alpha} + \frac{P_\alpha}{\varepsilon_\alpha}\frac{\mathrm{d}\tau}{\mathrm{d}P_\alpha} - \frac{P_\alpha \tau}{\varepsilon_\alpha^2}\frac{\mathrm{d}\varepsilon_\alpha}{\mathrm{d}\tau}\frac{\mathrm{d}\tau}{\mathrm{d}P_\alpha} = (\gamma_\alpha - 1)\left(1 - \frac{\gamma_\alpha}{\Gamma_\alpha}\right)\frac{1}{P_\alpha} \tag{9.4}$$

and the partial pressure increments as the explicit functions of the total pressure increment

$$\mathrm{d}P_\alpha = \frac{C_\alpha^2}{C^2}\mathrm{d}P, \tag{9.5}$$

where the squared Lagrangian speed of sound of a component is $C_\alpha^2 \equiv -\mathrm{d}P_\alpha/\mathrm{d}\tau = (\partial P_\alpha/\partial \varepsilon_\alpha)P_\alpha - \partial P_\alpha/\partial \tau$ and total pressure is $P(\varepsilon, \tau) = \sum_\alpha P_\alpha(\varepsilon_\alpha, \tau)$. In the computations, it is convenient to use the fraction of the specific energy of a component $\gamma_\alpha^\varepsilon \equiv \varepsilon_\alpha/\varepsilon$ (see [9])

$$\mathrm{d}\gamma_\alpha^\varepsilon = \gamma_\alpha^\varepsilon \frac{\gamma_\alpha - \gamma}{\Gamma}\frac{\mathrm{d}P}{P}, \tag{9.6}$$

where $\Gamma \equiv C^2\tau/P$, $\Gamma_\alpha \equiv C_\alpha^2\tau/P_\alpha$. The increment of the dimensionless variable $\gamma \equiv \sum_\alpha \gamma_\alpha \varepsilon_\alpha \big/ \sum_\alpha \varepsilon_\alpha$ can be evaluated (see [9])

$$\mathrm{d}\gamma = (\gamma - 1)\left(1 - \frac{\gamma}{\Gamma}\right)\frac{\mathrm{d}P}{P} \tag{9.7}$$

as expected for a single component (Eq. 9.4).

The assumption that the entropy variation that is negligibly small is used to compute only the variations in the dimensionless coefficients. The local model for EOS proposed resolves the uncertainty occurring when the specific internal energy and the pressure of a mixture component behind SW (rarefaction wave) are computed from known values of $\gamma_\alpha^\varepsilon \equiv \varepsilon_\alpha/\varepsilon$ behind the wave. Also the local EOS with the a priory known dimensionless coefficients as the functions of the full pressure behind the wave reduce to the Riemann problem solver to the case of one temperature gas with EOS of the ideal gas [9, 11, 18].

9.3 Shock Wave Structure in Hydrogen Plasma

As a test, an SW structure arising in hydrogen plasma in a tube at rest with a piston moving with the constant velocity into the gas is considering. If the hydrogen is completely ionized, the system involves protons and electrons. The “massless”

electrons transfer heat by conduction. Also electrons exchange energy with the protons. The temperatures of protons and electrons near SW are different. The kinetic coefficients are specified as by Shafranov [16] as:

$$\rho Q_p = \frac{3m_e}{m_p}\frac{n_e}{\tau_e}(kT_e - kT_i), \tag{9.8}$$

where

$$\tau_e = \frac{3\sqrt{m_e}(kT_e)^{3/2}}{4\sqrt{2\pi}\lambda q^4 n_e}, \tag{9.9}$$

the Coulomb logarithm

$$\lambda = \frac{1}{2}\ln\left(\frac{kT_eT_i}{T_e+T_i}\right)^3 \Big/ (q^6 n_e), \tag{9.10}$$

$n_e = n_i = \rho/m_p$, and the heat flux in electron heat conduction is $-\kappa_e \nabla(kT_e)$ with thermal conductivity

$$\kappa_e = 3.16 n_e k T_e \tau_e / m_e. \tag{9.11}$$

An ideal monoatomic gas for protons and electrons with $\Gamma_{i,e} = 5/3$ is used in EOS:

$$P_\alpha = \left(\frac{5}{3} - 1\right)\rho\varepsilon_\alpha\varepsilon_\alpha = \frac{k_{\mathbf{B}}T_\alpha}{(5/3-1)m_p}, \quad \alpha = p, e. \tag{9.12}$$

The initial hydrogen pressure is selected approximately equal to the atmospheric pressure. The initial temperature is chosen so as to achieve the ionization. The hydrogen is half ionized when the initial temperature is 10^4 K and the density is $10^{-6}\,\mathrm{g\,cm^{-3}}$ [19]. The Saha ionization equation for hydrogen

$$\frac{n_e n_p}{n_H} = \frac{(m_e kT)^{3/2}}{(2\pi\hbar)^3} g_e \exp(-I/(kT)) \tag{9.13}$$

with the ionization potential $I = 13.6\,\mathrm{eV}$ provides the ionized hydrogen at rather low temperature $kT \ll I$.

The electroneutrality condition is assumed to hold, the concentrations and velocities of the electrons and protons everywhere coincide, and the jump of charge on SW is neglected. Shafranov [16] worked with relations on discontinuities and solved a system of ODEs from both sides of the discontinuity. Due to the piecewise-smooth temperature of electrons $(-\partial T_e/\partial x)_1 > (-\partial T_e/\partial x)_2$ (Fig. 9.1), the heat fluxes are different on SW because of the independence of the conduction coefficient, κ_e (Eq. 9.11) from the concentration. Shafranov [16] did not take into account the

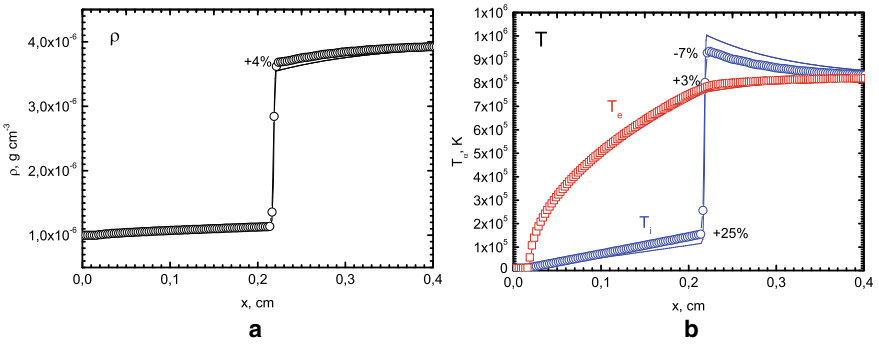


Fig. 9.1 Evaluation of: **a** density profiles, **b** proton temperatures profiles (circles) and electron temperature profiles (squares). Exact solution of Shafranov problem is given by solid curves for strong SW for $M = 16$ at time moment $6.65 \cdot 10^{-7}$ s. Numbers near SW on the plots show relative accuracy of numerical solution

heat conduction on SW. Only PDEs can operate with the different heat fluxes. The easiest way is to solve the system of PDEs in the Lagrangian coordinates for density, momentum, and specific energies of protons and electrons on fine computational grid, see details in [11]. To exclude discontinuities, one should introduce the viscosity protons passing from the hyperbolic system of equations to a parabolic one. In the present calculations, we are not interested in the fine structure of SW due to finite protons viscosity and artificially reduced the physical protons viscosity. The SW width is less than the heat conduction transfer region on the factor $\sqrt{m_e/m_p}$.

In the fixed Eulerian grid, the gas in the initial state $t = 0$ contains two constant states on the left and right sides from the contact discontinuity near the right boundary in the computational region ($0 < x < 20$ cm): $\rho_L = \rho_R = 10^{-6}\,\mathrm{g\,cm^{-3}}$, $T_L = T_R = 10^4$ K, $v_L = 0$, $v_R = -4 \times 10^7\,\mathrm{cm\,s^{-1}}$. As a result of the discontinuity decay, one has two strong shock waves moving to the left and right directions from the contact discontinuity. It is interesting to consider the left shock wave moving into a gas at rest. The velocity of the contact discontinuity is $v = -2 \times 10^7\,\mathrm{cm\,s^{-1}}$ in the coordinates' frame of gas at rest on the left side from the contact discontinuity. Thus, the task is equivalent to a piston moving with a velocity $v = -2 \times 10^7\,\mathrm{cm\,s^{-1}}$ into a gas at rest.

The velocity of a moving piston is chosen so as to obtain a strong stationary SW, on which the density jump near to $\rho_2/\rho_1 = 4$ for adiabatic index 5/3. The numerical solution of the problem is shown in Fig. 9.1. A numerical grid contains 8000 intervals on the region 20 cm. In the direction of motion of the gas, a stationary SW is formed, propagating relative to the unperturbed gas. The profiles of all quantities near the SW shift at a constant velocity and remain unchanged. The plasma is in a nonequilibrium state near SW, but equilibrium is established at some distance behind the SW front. Jumps in the proton density and temperature are observed on SW. Due to electron heat conduction, the electron temperature T_e is continuous and piecewise-smooth. Figure 9.1 demonstrates good agreement of the numerical solution with the "exact" solution obtained in the Lagrangian coordinates.

Behind SW, the gas velocity is equaled -2×10^7 cm s^{-1}, the pressure is equaled 5.37×10^8 din cm^{-2}, and the temperature is equaled 8.11×10^5 K. The density behind SW is equaled 4.00×10^{-6} g cm^{-3}. The simplest way to define the SW velocity D is the constant mass flux on both surfaces of SW $(0 - D) \times 10^{-6}$ g cm^{-3} = $(2 \times 10^7$ cm s$^{-1} - D) \times 3.955 \times 10^{-6}$ g cm^{-3}. Thus, $D = -2.68 \times 10^7$ cm s^{-1}, and its Mach number is $M = |D|/c_s = 16$, where c_s is the sound speed, relative to the gas at rest.

9.4 Taking into Account Radiation Effects

In the previous Sect. 9.3, some contradictory results were obtained. The obtained pressure is equaled 5.37×10^8 din cm^{-2}, and the temperature is equaled 8.11×10^5 K behind SW. The blackbody radiation pressure with such temperature is defined as:

$$P_\gamma = \left(\frac{4}{3} - 1\right)\rho\varepsilon_\gamma = \left(\frac{4}{3} - 1\right)aT_\gamma^4 = 1.2 \times 10^9 \,\text{dyn cm}^{-2}, \tag{9.14}$$

where a radiation constant is $a = \pi^2 k_\text{B}^4 \big/ \left(60(\hbar c)^3\right)$. The neglected radiation pressure exceeds the gas pressure. Therefore, for a correct physical formulation of the problem, one needs to introduce photons and take into account their energy transfer and energy exchange with electrons.

In pure protons-electrons system, one has the photons interacting with electrons in Compton scattering. For estimates, one can accept the constant Thomson cross-section $\sigma_\mathbf{T} = 8\pi r_e^2/3 = 6.65 \times 10^{-25}$ cm^2 for interactions. Then the free path of photons is estimated as:

$$(\sigma_\mathbf{T} n_e)^{-1} \sim 10^6 \,\text{cm}. \tag{9.15}$$

Such space scale exceeds the hydrodynamic region for the steady SW formation lesser 1 cm. The timescale for the energy exchange between electrons and photons is another parameter:

$$\tau = \frac{1}{c\sigma_\mathbf{T} n_e} \sim 10^{-4} \,\text{s}. \tag{9.16}$$

This parameter is larger than the hydrodynamic time 10^{-6} s from the previous section.

Therefore, the solution from the previous section can be a quasi-stationary at some times scales and space scales. To check it, one can formulate the problem for one temperature of protons and electrons $T_p = T_e$ and for the different temperature of radiation T_γ:

$$\rho\varepsilon_\gamma = aT_\gamma^4,\ P_\gamma = \left(\frac{4}{3} - 1\right)\rho\varepsilon_\gamma. \tag{9.17}$$

It is possible to disregard electron heat conduction and introduce the equation of transport for the radiation energy density as:

$$\frac{\partial\rho\varepsilon_\gamma}{\partial t} + \mathbf{v}\nabla\left(\rho\varepsilon_\gamma\right) = \operatorname{div}\mathbf{F}_\gamma - \rho Q_{p,e}, \tag{9.18}$$

where the flux is determined by the gradient of the zeroth moment of the photons distribution function. In the opaque case, $\mathbf{F}_\gamma^{\text{thick}} = -(\operatorname{grad}\rho\varepsilon_\gamma)/(3c\sigma_{\mathbf{T}}n_e)$, and in the transparent case $F_\gamma^{\max} = c\rho\varepsilon_\gamma$. In the arbitrary case, we can use the interpolation (the flux limiter):

$$\mathbf{F}_\gamma = \frac{\mathbf{F}_\gamma^{\text{thick}}}{\left|\mathbf{F}_\gamma^{\text{thick}}\right|/F_\gamma^{\max} + 1}. \tag{9.19}$$

The constraint on the flux refers to the introduction of nonlinear thermal conductivity and a certain arbitrary fit of the fluxes in the intermediate case. The nonlinear diffusion flux transfers the parabolic equation in the opaque region into hyperbolic transport equation in the transparent region for the spectral energy densities.

The exchange of energy between radiation and matter is described by the relaxation to a thermal distribution:

$$\rho Q_{p,e} = c\sigma_{\mathbf{T}}n_e\left(\rho\varepsilon_\gamma - \rho\varepsilon_\gamma^{\text{th}}\right), \tag{9.20}$$

where equilibrium blackbody radiation energy density $\rho\varepsilon_\gamma^{\text{th}}$ should be calculated for the total energy density of radiation and matter $\rho\varepsilon + \rho\varepsilon_\gamma$.

Figure 9.2 illustrates the evaluation of profiles of the density ρ and the temperatures of gas and radiation $T_{p,e}, T_\gamma$ with time. At the first time moment $t = 1.70{\cdot}10^{-5}$ s, radiation is negligible, and its role is unimportant. The density and temperature profiles look as in the previous section for the hydrodynamic case. In the next time moments 1.12×10^{-5} s and 1.70×10^{-5} s, radiation is still small due to a matter transparency, but radiation plays considerable role in the energy losses in the relaxation zone behind SW. The maximum density becomes larger than 4×10^{-6} g cm^{-3} in the relaxation zone. At time moment more than 10^{-5} s, the SW structure obtained in the previous section changes. The density jump is still 4, but the gas temperature decreases value 8×10^5 K. At time moment more than 10^{-5} s, the SW structure obtained in the previous section becomes inapplicable.

The steady solution with taking into account radiation forms at SW propagation on the distance $\sim 10^8$ cm during 5 s, see Fig. 9.3. This steady solution contains nonequilibrium radiation (the space scale in Fig. 9.3 cannot resolve nonequilibrium region near SW), slightly preheated gas before SW, and relaxation zone after SW (the zone is not resolved in Fig. 9.3). In the relaxation zone, radiation achieves the thermal equilibrium with matter $T_\gamma = T_{p,e}$, but the radiation energy flux from the SW

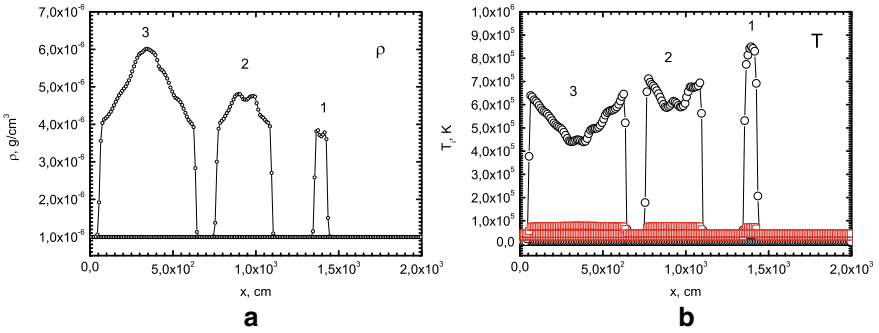


Fig. 9.2 Evaluation of profiles of: **a** density, **b** temperatures of gas and radiation. Density profiles, proton-electrons temperature profiles (circles), radiation temperature profiles (squares) at the contact discontinuity decay with taking into account radiation. The gas velocity near the right boundary is equaled $v = -4 \cdot 10^7 \text{cm} \cdot \text{s}^{-1}$ and Mach number is equaled $M = 16$. Numbers indicate the time moments: $5.36 \cdot 10^{-6}$ s (Eq. 9.1), $1.12 \cdot 10^{-5}$ s (Eq. 9.2), and $1.70 \cdot 10^{-5}$ s (Eq. 9.3)

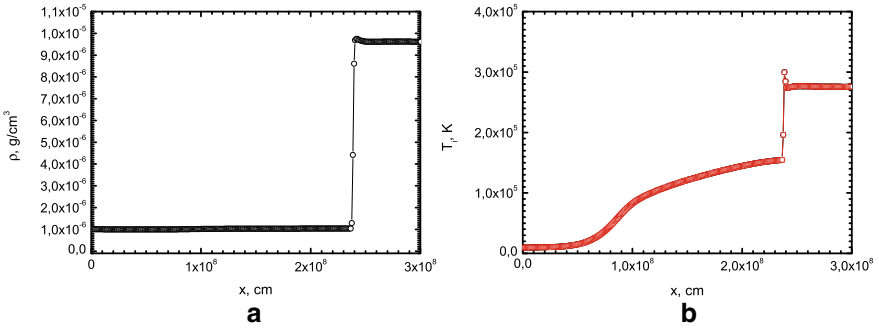


Fig. 9.3 Steady solution of: **a** density profile, **b** proton-electrons temperature profile (circles) and radiation temperature profile (squares) at SW with taking into account radiation at large time moment more than 5 s. Used computational grid and the space scale cannot resolve nonequilibrium region near SW with the size $\sim 10^4$ cm. The jump of the radiation temperature on SW is an artifact of the insufficient resolution of the grid

zone is considerable. The density after SW achieves the value of about 10^{-5} g cm^{-3} due to the radiation energy loss. The total adiabatic index for protons, electrons, and radiation is less than value 5/3 for one atomic gas due to the considerable role of radiation in the total energy.

The radiation flux plays considerable role not only in the relaxation zone. To obtain a fine structure of the stationary SW on the distance ~1 mm at time moment more than 10^{-5} s, one should take into account radiation fluxes.

9.5 Conclusions

The proposed approximate Riemann problem solver yields the qualitatively and also quantitatively good results despite the assumption for the entropy jump smallness used in computation of the dimensionless coefficients of the gases. It is shown that the steady solution with the fine structure (~1 mm) of the strong SW in hydrogen plasma with Mach number $M = 16$ is quasi-stationary on the distance ~50 cm during the time scale ~10^{-5} s. To prove this result, it was necessary to introduce the radiation component of the plasma, and thus, it was necessary to transfer from the ideal gas low to a general equation of state.

Acknowledgements We are grateful to the referees for their careful reading of the manuscript, valuable suggestions, and remarks.

References

1. Basko, M.M., Churazov, M.D., Aksenov, A.G.: Prospects of heavy ion fusion in cylindrical geometry. Laser Part. Beams **20**, 411–414 (2002)
2. Anisimov, S.I., Zhakhovskiĭ, V.V., Inogamov, N.A., Nishihara, K., Petrov, Y.V., Khokhlov, V.A.: Ablated matter expansion and crater formation under the action of ultrashort laser pulse. JETP **103**, 183–197 (2006)
3. Fortov, V.E., Hoffmann, D.H., Sharkov, B.Y.: Reviews of topical problems: intense ion beams for generating extreme states of matter. Phys. Uspekhi **51**, 109–131 (2008) (in Russian)
4. Bruenn, S.W.: Stellar core collapse—numerical model and in fall epoch. ApSS **58**, 771–841 (1985)
5. Pelanti, M., Shyue, K.-M.: A mixture-energy-consistent six-equation two-phase numerical model for fluids with interfaces, cavitation and evaporation waves. J. Comput. Phys. **259**, 331–357 (2014)
6. Zhukov, V.T., Zabrodin, A.V., Feodoritova, O.B.: A method for solving two-dimensional equations of heat-conducting gas dynamics in domains of complex configurations. J. Comput. Math. Math. Phys. **33**, 1240–1250 (1993) (in Russian)
7. Miller, G.H., Puckett, E.G.: A high-order Godunov method for multiple condensed phases. J. Comput. Phys. **128**, 134–164 (1996)
8. Dolence, J.C., Burrows, A., Zhang, W.: Two-dimensional core-collapse supernova models with multi-dimensional transport. Astrophys. J. **800**(10), 1–14 (2015)
9. Aksenov, A.G.: Computation of shock waves in plasma. J. Comput. Math. Math. Phys. **55**, 1752–1769 (2015)
10. Vereshchagin, G.V., Aksenov, A.G.: Relativistic Kinetic Theory with Applications in Astrophysics and Cosmology. Cambridge University Press, Cambridge (2017)
11. Aksenov, A.G., Chechetkin, V.M., Tishkin, V.F.: Godunov type method and the Shafranov's task for multi-temperature plasma. Math. Models Comput. Simul. **11**, 360–373 (2019)
12. Colella, P., Woodward, P.R.: The piecewise parabolic method (PPM) for gas dynamical simulations. J. Comput. Phys. **54**, 174–201 (1984)
13. Aksenov, A.G., Churazov, M.D.: Deuterium targets and the MDMT code. Laser Part. Beams **21**, 81–84 (2003)
14. Aksenov, A.G., Chechetkin, V.M.: Supernova explosion mechanism with the neutrinos and the collapse of the rotation core. Astron. Rep. **62**, 834–839 (2018)

15. Aksenov, A.G., Chechetkin, V.M.: Large-scale instability during gravitational collapse and the escaping neutrino spectrum during a supernova explosion. Astron. Rep. **63**, 900–909 (2019)
16. Shafranov, V.D.: The structure of shock waves in a plasma. Sov. Phys. JETP **5**, 1183–1188 (1957)
17. Gear, C.W.: Numerical Initial Value Problems in Ordinary Differential Equations. Prentice-Hall, Inc., Englewood Cliffs, New Jersey (1971)
18. Colella, P., Glaz, H.M.: Efficient solution algorithms for the Riemann problem for real gases. J. Comput. Phys. **59**, 264–289 (1985)
19. Zeldovich, Y., Novikov, I.: Relativistic Astrophysics, vol. 1: Stars and Relativity. University of Chicago Press, Chicago (2017)

Chapter 10
Thruster Rotation Angle Control During Contactless Removal of Space Debris Objects

Vladimir A. Obukhov, Alexander I. Pokryshkin, and Victoria V. Svotina

Abstract The chapter deals with the issues of contactless removal of space debris objects, the orbit of which is changed by a high-velocity ion beam injected from the service spacecraft moving in the immediate vicinity of the debris object. We consider the issues of controlling the angles of rotation of electric propulsion thrusters to implement changes in the thrust components of electric propulsion system in the longitudinal and transverse directions required during the debris object transportation. Arrangement of thrusters is proposed taking into account the location of solar arrays and the difference in permissible angles of thruster deflection in different planes. We analyze possible options for the thruster rotation angles to provide the required values of the thrust projection onto the axes of the spacecraft-associated coordinate system. We propose an algorithm for controlling the thruster rotation angles to implement the required thrust projection values, which allows to control the sign of the momentum relative to the spacecraft longitudinal axis depending on the accumulated total momentum. The results of modeling the electric propulsion system operation during the space debris objects transportation are presented.

V. A. Obukhov (✉) · A. I. Pokryshkin · V. V. Svotina
Research Institute of Applied Mechanics and Electrodynamics of the Moscow Aviation Institute (RIAME MAI), 5 Leningradskoye shosse, Moscow 125080, Russian Federation
e-mail: deutscher@mail.ru

A. I. Pokryshkin
e-mail: flox_ap@mail.ru

V. V. Svotina
e-mail: vsvotina@mail.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_10

10.1 Introduction

Much attention has recently been paid to the removal of space debris object (SDO) from near-Earth space. Many different methods of removal of large objects on the disposal orbits or low orbits for their destruction in dense layers of the Earth's atmosphere are proposed. The concept of contactless removal of SDO (so-called Ion Shepherd technology) was proposed in [1], according to which the SDO orbit is altered by a high-velocity ion beam injected from a service spacecraft (SSC) moving in immediate vicinity of SDO. There are quite a few publications that address some or other issues of controlling the SSC-SDO cluster during the contactless removal. In [2, 3], such issues were considered in a broad sense; however, they did not take into account the peculiarities of the SSC design.

Within the framework of the Ion Shepherd concept, the SSC scheme was proposed in [4], in which electric propulsion system (EPS) comprising two electric propulsion thrusters (EPTs), each of which is mounted on a two-coordinate gimbal, is used to produce thrust that should compensate the thrust of a high-velocity ion beam source. In this case, ion beam source (IBS) and EPS are mounted along the SSC longitudinal axis.

For such a spacecraft, the strategy and algorithms of controlling the motion of the SSC center of mass by creating control impacts in the plane orthogonal to the SSC longitudinal axis were considered in [5]. For numerical modeling and performance analysis of the considered control algorithms, a simplified model of ion beam impact on SDO was used in [5].

In this chapter, we consider the problems of controlling the angles of the EPT rotation to implement changes in the EPS thrust vector components in the direction of the SSC longitudinal axis and in the transverse direction. Motion control for the SSC-SDO cluster, in addition to its lateral motion control, also requires to control the thrust projection onto the SSC longitudinal axis. Algorithm of thruster rotation angle control for implementing the required values of the thrust projections that allows to control the sign of the momentum relative to the SSC longitudinal axis depending on the accumulated total momentum should be based on a certain arrangement of thrusters, taking into account the solar arrays location and the difference in permissible thruster deflection angles in different planes. It should be noted that pivoted thrusters can be used not only to create the required values of thrust projection onto the axes of the associated coordinate system, but also to unload the flywheels of the SSC inertial attitude control system.

Before a description of the thruster rotation angle control, a problem statement with service spacecraft is described in Sect. 10.2. The ideas of the thruster rotation angle control are introduced in Sect. 10.3. The ion beam momentum transfer modeling is presented in Sect. 10.4. The results of spacecraft motion dynamics simulation are presented in Sect. 10.5. Finally, the conclusions of the study are reported in Sect. 10.6.

10.2 Problem Statement: Service Spacecraft Design

The problem of controlling the EPT rotation angles is considered for a given SSC design [4]. Figure 10.1 shows the SSC design and axes of the SSC-associated coordinate system. IBS has constant thrust and is stationary mounted on the SSC longitudinal axis. The EPT pair, which compensates the IBS thrust and ensures the SSC-SDO cluster motion, is also located on the SSC longitudinal axis, but from the opposite side of IBS. Nominal thrust of EPS is directed along the SSC longitudinal axis. The EPT thrust is constant. At the same time, the thrusters can rotate relative to two axes changing the direction of the thrust, which leads to the appearance of lateral thrust projections, as well as to the shortening of the thrust projection onto the SSC longitudinal axis as compared to its nominal value.

In this chapter, it is assumed that the SSC orientation is controlled by the onboard attitude control system. The main problem in this case consists, on the one hand, in keeping the IBS axis in the direction to SDO and, on the other hand, in ensuring the orientation of SA toward the Sun. We consider the SA axis location in the local horizontal plane. It is assumed that at the stage of the SDO transportation from geostationary orbit (GEO) region, the SSC center of mass motion control system provides a control for the SSC lateral motion and for the relative distance between SSC and SDO.

The SSC lateral motion control strategy assumes that, due to the EPT rotation, SSC should be shifted in such a way that the vector of relative distance between SSC and SDO coincides with the transversal direction of the orbital coordinate system [5].

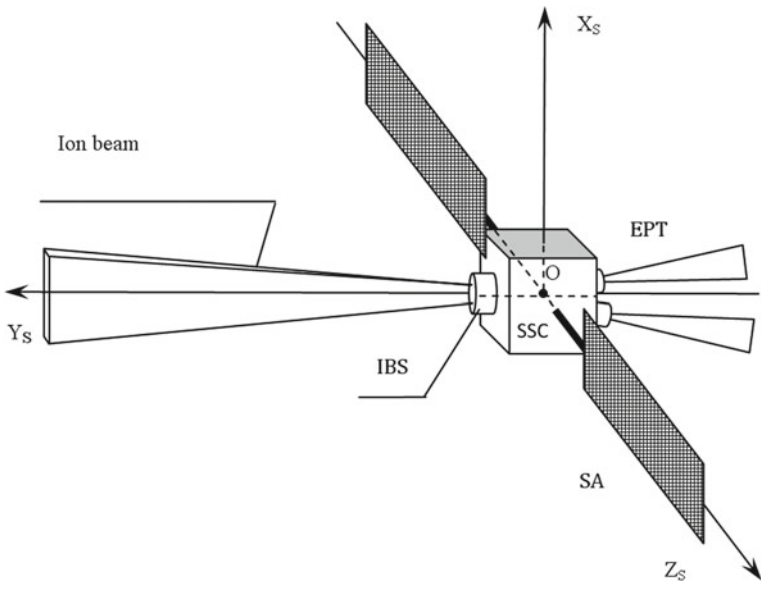


Fig. 10.1 SSC structural diagram and axes of the SSC-associated coordinate system

Angles between the direction of the transversal and the axes of the SSC-associated coordinate system being orthogonal to the longitudinal axis are considered as parameters for controlling the SSC lateral motion. When controlling the SSC-SDO relative distance, a proportionally differentiating controller is applied that uses data on the difference of the SSC-SDO relative distance from a certain average value, as well as on the derivatives of the change in the relative distance. The lateral motion and relative distance are controlled by the EPT rotation in two planes.

Turns of the EPS thrusters create momentums relative to the SSC-associated axes. It is assumed that such momentums are compensated by the flywheels. On the other hand, pivoted thrusters of EPS can in turn be used to unload the flywheels.

The aim of the work was to select the EPT arrangement on board SSC and develop an algorithm for the EPT rotation control, which would allow for control actions during the SDO transportation. The performance analysis of the EPT rotation control algorithm was carried out by numerical modeling using a simplified model of IB impact on SDO for different values of its parameters and algorithms of the SSC-SDO cluster motion control.

10.3 Electric Propulsion Thruster Angle Control

Before a description of the thruster rotation angle control, the thruster layout onto spacecraft-associated coordinate system is presented in Sect. 10.3.1. The thruster rotation control algorithm is described in Sect. 10.3.2.

10.3.1 The Thruster Layout. Thrust Projections onto the Axes of SSC-Associated Coordinate System

In this chapter, it is assumed that the SA axis is parallel to the axis OZ_S of the SSC-associated coordinate system. It is assumed that the ranges of the EPT turn angles in different planes may vary. In order to reduce the influence of the plasma plumes of EPS on SA, the thrusters are located in OXY plane, orthogonally to the SA rotation axis. With this, a larger range of angles of rotation is in OXY plane relative to the axis Z. The first EPT rotation is carried out by an angle δ_Z, while the second EPT rotation is implemented by an angle δ_X. The equations for projections of the relative thrust of two EPT onto the SSC-associated axes are provided by Eq. 10.1.

$$\begin{aligned} p_X &= -\cos\delta_{X1} \times \sin\delta_{Z1} - \cos\delta_{X2} \times \sin\delta_{Z2} \\ p_Y &= \cos\delta_{X1} \times \cos\delta_{Z1} + \cos\delta_{X2} \times \cos\delta_{Z2} \\ p_Z &= \sin\delta_{X1} + \sin\delta_{X2} \end{aligned} \tag{10.1}$$

Here, the projections of the relative EPS thrust are the ratio of the sum of projections of the EPT thrusts to the thrust of a single thruster P_E in the form of Eq. 10.2.

$$p_i = (P_{iS1} + P_{iS2}) P_E^{-1} \tag{10.2}$$

Equation 10.1 serves as the basis for determining the EPT rotation angles providing the required values of the thrust projections onto the axes of the associated coordinate system. Note that the number of unrestricted variables in these equations is by one more than equations.

Of the possible variants of the thruster rotation angles with respect to axis OZ_S, we exclude the case when the plumes of the operating thrusters are directed toward each other. Variants of the EPT rotation angles with respect to axis OX_S make it possible to control the sign of the momentum relative to axis OY_S. This should be taken into account in the EPT rotation control algorithm during the accumulation of the total momentum relative to axis OY_S for unloading the flywheels.

10.3.2 The Thruster Rotation Control Algorithm

The following algorithm for determining the EPT rotation angles is proposed.

Step 1. Using the algorithms for lateral motion control and longitudinal thrust component control, the required values of the relative thrust projections of two EPT on the axes of the SSC-associated coordinate system are determined.

Step 2. Using Eq. 10.1, the maximum and minimum acceptable values of the EPS thrust projection onto the longitudinal axis OY_S are determined taking into account the implementation of lateral control as well as the sign of the total momentum relative to the axis OY_S. If the required value of the thrust projection onto the longitudinal axis exceeds the permissible range, the control corresponding to such boundary values is implemented.

Step 3. If the required value of the thrust projection onto the longitudinal axis is within the acceptable range, then the control is provided as follows.

- Step 3.1. The EPT rotation angles relative to the axis OX_S are taken equal to each other and equal to the values corresponding to the required values of the thrust projection on the axis OZ_S. The EPT rotation angles with respect to the axis OZ_S are calculated on the basis of Eq. 10.1 taking into account the required values of the EPS thrust projection onto the axes OX_S and OY_S.
- Step 3.2. If one of the calculated thruster rotation angles with respect to axis OZ_S exceeds the permissible deflection value, then for the wider calculated rotation angle in terms of its absolute value, the rotation angle value is assumed as the maximum permissible

value. The second rotation angle relative to the axis OZ_S and the rotation angles relative to the axis OX_S are calculated by Eq. 10.1 based on the required values of the thrust projections, taking into account the sign of the total momentum relative to the axis OY_S.

10.4 Model of the Ion Beam Momentum Transfer to Space Debris Object

Calculation of the IB momentum transfer to SDO is an independent complicated problem. The forces and momentums acting on SDO depend on the SDO configuration, the location and orientation of SDO relative to IB, and also on the IB parameters. Various aspects of the calculation of forces and momentums are considered in [2, 3, 6, 7]. When analyzing divergence of IB flowing out into the outer space, it is necessary to take into account the initial divergence angle and the action of electron pressure and ambipolar electric field in the beam. For a conical IB, the end formulas were obtained, which allow one to calculate the parameters of IB in the far field in the region of interaction with SDO [8]. When calculating the force acting on SDO, the shape of the latter is usually idealized, taking it for a sphere or a cylinder. In the exact calculation of forces and momentums, to assess the quality of the control process, it is necessary to conduct statistical modeling taking into account the angular motion of SDO, in which the initial conditions for the SDO orientation take random values. In this chapter, we have interest to evaluate algorithms of the SSC control and the EPT rotation control for various types and different parameters of the SDO angular motion. To this end, it is proposed to use a simplified simulation model of IB impact on SDO, presented in [5]. The model assumes that for a specific SDO orientation, there is a circle with the effective radius R_T, the influence of IB on which is equivalent to IB impact on SDO.

Since SDO rotates generally, it is proposed in the simulation model to replace the real object with a circle of effective radius, which changes its size from the maximum value to the minimum value according to the harmonic law in the form of Eq. 10.3.

$$R_T = R_{\mathrm{MAX}}(1 - k_R) + R_{\mathrm{MAX}} \times k_R \times \sin\left(\varphi_R + \frac{2\pi}{T_R} \times t\right) \tag{10.3}$$

Here, k_R defines the relative amplitude of the change in the effective radius, and φ_R, T_R are the phase shift and period of the oscillatory component for the effective radius, respectively.

In [9], the results of a study of the model of wedge-shaped ion beam injector are presented; such injector differs by narrow initial divergence angles of less than 2° and 4° in two mutually perpendicular directions. IB with such initial characteristics is effective when acting on SDO from the distance of 20 m. The magnitude of the force P_{TN} acting on SDO in the direction of relative range and depending on the

thrust P_I generated by IB, when the entire wedge-shaped beam reaches SDO, on the relative range L and the IB divergence angle β_I is taken as Eq. 10.4.

$$P_{\text{TN}} = \begin{cases} P_I & R_T \leq L \times \text{tg}(\beta_I) \\ \frac{P_I \times R_T}{L \times \text{tg}(\beta_I)} & R_T > L \times \text{tg}(\beta_I) \end{cases} \tag{10.4}$$

Due to a certain degree of approximation of the proposed dependences, it is assumed here that IB has a uniform angular distribution of ion flow density.

During the SDO rotation due to various factors, the components of the force of the IB impact on SDO arise in a plane orthogonal to the relative range direction. In the simulation model, it is proposed to use a harmonic law to describe these forces also, and the magnitude of such forces is assumed to be proportional to the force acting in the direction of relative range. The components of the lateral forces are defined by Eq. 10.5.

$$P_{TVi} = P_{\text{TN}} \times k_{Vi} \times \sin\left(\phi_{Vi} + \frac{2\pi}{T_{Vi}} \times t\right) \tag{10.5}$$

Here, i designates the values of X and Z, k_{Vi} is the ratio of the maximum value of the lateral force component to the magnitude of the force acting in the direction of relative range, and ϕ_{Vi}, T_{Vi} are the phase shift and the period of the oscillatory lateral force of IB impact on SDO.

10.5 Simulation of Spacecraft Motion Dynamics

For spacecraft motion dynamics simulation, spacecraft motion equations and their integration results are presented in Sect. 10.5.1. The dynamics of spacecraft-SDO relative distance is described in Sect. 10.5.2. The angles of the thruster rotation examples in SDO removal are demonstrated in Sect. 10.5.3.

10.5.1 Integration of Motion Equations

Numerical simulation was performed to assess the possibility of controlling SDO transportation into the disposal orbit using the pivoted thrusters of EPS. The equations of the SSC motion and SDO motion relative to SSC were considered in a geocentric inertial coordinate system without taking into account disturbances from the non-sphericity of the Earth and disturbing factors of a higher order of smallness. The integration duration was up to 3 days.

The following data were used. The initial SDO position was set by the parameters of the elliptical orbit: the semimajor axis 42,157 km, the eccentricity 0.0005, and

the inclination 0°. SSC moves in an orbit coinciding with that of SDO. SSC is 40 m behind SDO, 5 m in vertical deflection, and 5 m away from the orbit plane. The SDO mass is 2000 kg, and SSC initial mass is 1500 kg. The ion beam injector thrust is 50 mN, and the propulsion system comprises two EPTs with 35 mN thrust each.

As an algorithm for controlling the SSC lateral motion, we consider the algorithm of angular deflections of EPS thrust vector direction taking into account the magnitudes and derivatives of the angles between the transversal and the axes of the SSC-associated coordinate system being orthogonal to the SSC longitudinal axis [5] with the parameters $k_0 = 0.3$, $k_1 = 300$ s. The value of control angle is limited by $\alpha_i \leq \alpha_{\max} = 5°$.

The control options with and without taking into account the control algorithm for the thrust vector longitudinal component are considered. As an algorithm for controlling the thrust vector longitudinal component, we considered a proportional-derivative controller with parameters $k_{0D} = 0.0001, k_{1D} = 0.1$ s; the average relative range took the values: 45, 30, and 20 m. We took into account the restriction of the total momentum relative to the SSC longitudinal axis to ± 10 Nm s. When calculating the momentums produced by EPS relative to the center of the SSC mass, the deflections of the EPT position from the origin of the SSC-associated coordinate system were assumed to be 1 m. The amplitudes and phase shifts for the oscillatory component of the force of IB impact on SDO for the axes OX_S, OY_S, and OZ_S were, respectively: 0.1, 90°; 0.3, 0°; 0.1, –90°. The oscillation periods were equal along all axes and amounted to 10 min, 1 h, and 3 h. The maximum effective radius was 1.5 m.

10.5.2 Dynamics of Changes in the Relative Range Projection onto the Transversal

The dependence of the relative range projection onto the transversal most vividly characterizes the process of the SDO removal from the GEO region. Figure 10.2 shows the simulation results for different parameters of the control algorithms and different parameters of the simulation model of the IB momentum transfer to SDO.

It is obvious from the graphs in Fig. 10.2 that show the changes in the relative range projection onto the transversal that the control of only lateral deviations for the rotation period of 1 h or less is enough to transfer SDO to the disposal orbit. In this case, the thrust vector longitudinal component is the maximum possible and the increment in the SDO orbit altitude during 3 days is large enough—of about 90 km. At the same time, with a rotation period of 3 h, the SDO removal becomes impossible, the SDO acceleration has low value for a long time, and SSC overtakes SDO.

The use of control for the thrust vector longitudinal component together with the control for the lateral deviations with an average relative distance of 45 m makes it possible to maintain a predetermined distance between objects. This is achieved by reducing the EPS thrust vector longitudinal component through the EPT pivoting.

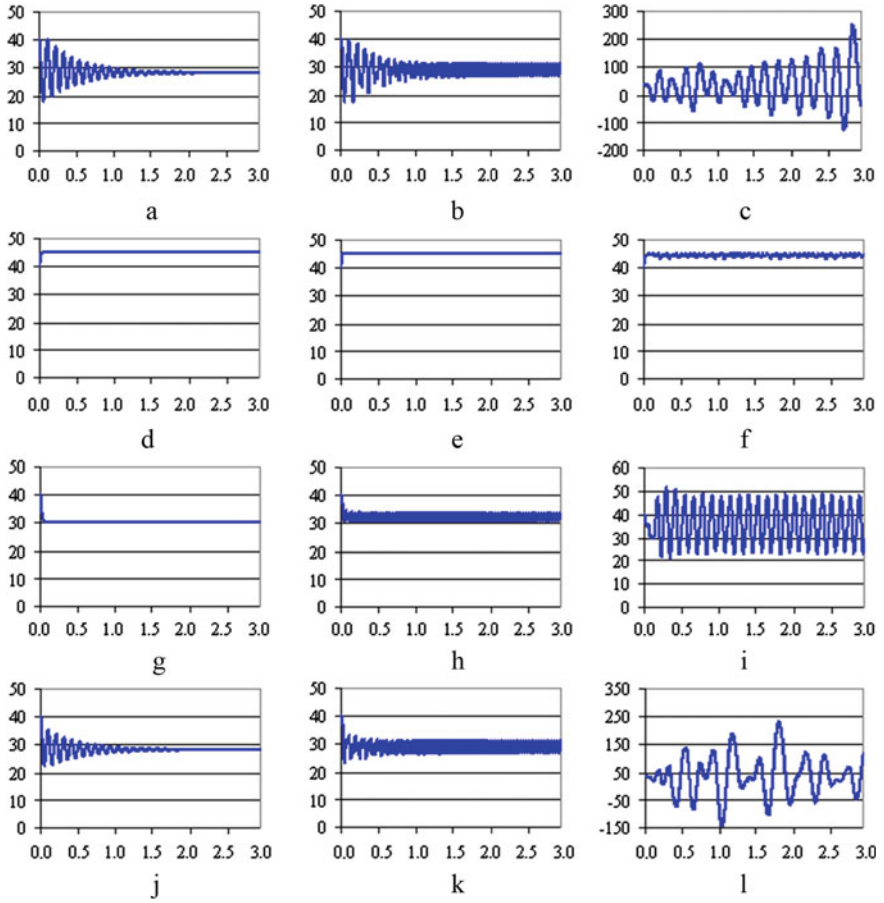


Fig. 10.2 Projections of relative distance (m) onto the transversal depending on time (day): **a**–**c** control for lateral displacements only. The average distance when controlling the longitudinal component of the thrust vector: **d**–**f** 45 m, **g**–**i** 30 m, **j**–**i** 20 m. Period of the oscillation: **a**, **d**, **g**, **j** 10 min, **b**, **e**, **h**, **k** 1 h, **c**, **f**, **i**, **l** 3 h

As a result, the increment in orbit altitude is 60 km only. With the relative range reduction down to 30 m for the periods of 10 min, 1 h, and 3 h, it is possible to raise the orbit for 90, 85, and 70 km, respectively. With an average relative range of 20 m, the orbit can be raised for 95 km. At the same time, for the period of 3 h, the process of the SDO removal becomes impossible.

It can be seen from the below graphs that the smaller distance between the objects, the higher thrust impulse is transferred to SDO, which reduces the time of the SDO removal from the GEO region. If the average range is 20 m, we obtain the average relative range projection on the transversal of about 30 m. This is due to limitations on the acceleration magnitude, which can be realized using EPS considered in the

calculations. To maintain an average distance of 20 m, a thrust is required that is higher than that produced by the considered EPS.

Using the simulation model for the IB impact on SDO, one can conduct qualitative analysis for the process of the SDO removal from GEO for various values of the parameters of control algorithms, simulation model, and SSC. The dependencies shown in Fig. 10.2 were obtained using the algorithm for controlling the EPT rotation that was considered in this chapter.

10.5.3 Parameters of Spacecraft Motion Dynamics in the Process of SDO Removal from GEO

In Figs. 10.3, 10.4, 10.5 and 10.6 showing the changes in the parameters of the SSC motion dynamics during the SDO removal from the GEO region, the average range is 30 m, and the period of the oscillatory component of the effective radius of the IB

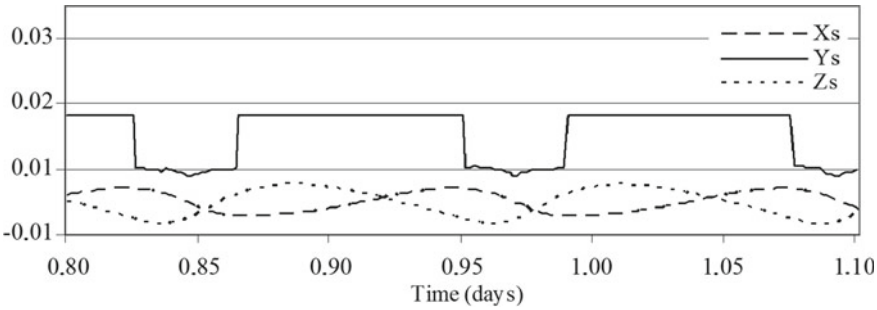


Fig. 10.3 Projections of the SSC acceleration (mm/s^2)

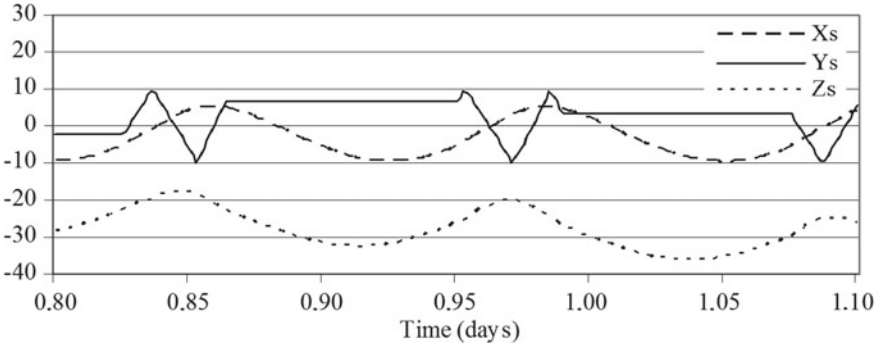


Fig. 10.4 Projections of total momentum (Nm s)

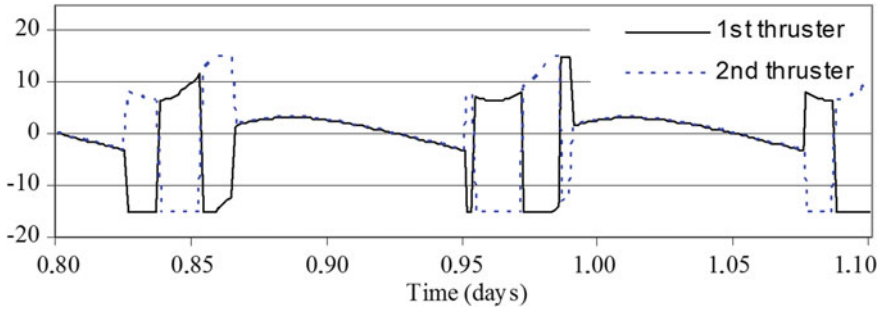


Fig. 10.5 Angles of the EPS thruster rotation relative to the axis OX_S (degree)

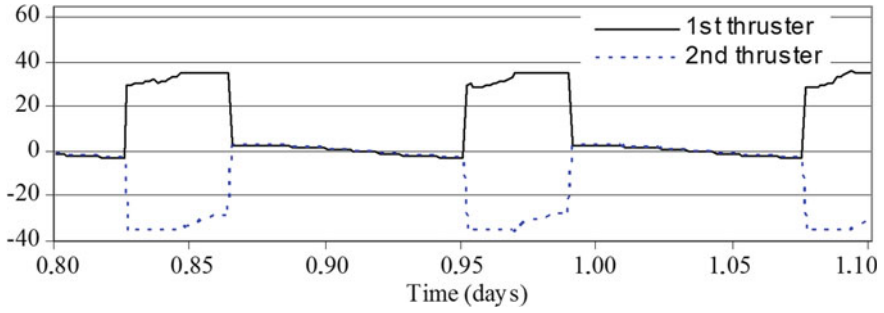


Fig. 10.6 Angles of the EPS thruster rotation relative to the axis OZ_S (degree)

momentum transfer is 3 h. The origin of curves corresponds to the flight duration of 0.8 days. The time is relative, and the dimension is min.

Figures 10.3, 10.4, 10.5 and 10.6 show that the changes in the SSC motion dynamics in the process of the SDO removal from the GEO region are periodic in nature, due to the oscillatory component of the force of the IB impact on SDO, with the period of 3 h.

The SSC acceleration projections onto the axes OX_S and OZ_S are smooth. It follows from the analysis for acceleration projection onto the axis OY_S that for maintaining average range of 30 m, the control should include sections with the maximum and minimum thrust projections onto the SSC longitudinal axis. In sections with maximum thrust projection, only the lateral SSC motion is controlled.

The changes in the projections of the total momentums onto the axes OX_S and OZ_S are smooth; the difference between its maximum and minimum values in the time interval of 0.8–1.1 days is about 18 Nm s. For the total momentum relative to the axis OY_S, when ± 10 Nm s is reached, the sign of the momentum produced by EPT relative to the axis OX_S changes. If the momentum sign is not changed, then the total momentum will be accumulated, and a large consumption of the propellant will be required for unloading the flywheels. It should be noted that the momentum sign change is provided by pivoting the thrusters about the axis OX_S.

The EPT rotation angles with respect to the axis OX_S are within $\pm 15°$. At the moments of change in the momentum sign relative to the axis OY_S, which are defined by the achievement of ± 10 Nm s by the total momentum projection, the signs of the EPT rotation angles change.

Changes in the EPT rotation angles with respect to the axis OZ_S are within $\pm 35°$. When the momentum relative to the axis OY_S changes its sign, which is defined by the achievement of ± 10 Nm s by the integral momentum, due to the change in the EPT rotation angles sign with respect to the axis OX_S, there are small jumps for one of the angles in the graphs for the rotation angle changes relative to the axis OZ_S. In the above graphs, the second angle in this case takes the maximum allowable value.

In the graphs for the EPT rotation angle changes, there are sections, in which the EPT rotation angles are small in magnitude and equal to each other. In these sections, the lateral SSC motion is controlled only.

The performed numerical simulation demonstrates efficiency of the proposed methodology for assessing reliability of the considered algorithms for controlling the EPT rotation in the process of the SDO removal from the GEO region.

The possibility to vary parameters of the simulation model for the IB momentum transfer to SDO allows us to analyze a wide range of different conditions for the ion beam impact on SDO.

10.6 Conclusions

The problem of contactless space debris removal is considered, in which the orbit is changed using a high-velocity ion beam injected from SSC of a given design, following the space debris object in its immediate vicinity.

An arrangement for the SSC electric propulsion thrusters is proposed taking into account various values of permissible thruster rotation angles in two planes, as well as the arrangement of solar panels. An algorithm is proposed for controlling the rotation of the electric propulsion system thrusters, taking into account the required thrust vector projection values in the lateral and longitudinal directions, as well as the value of the total momentum accumulated relative to the SSC longitudinal axis.

The simulation results demonstrate the efficiency of the proposed algorithm for controlling the EPT rotation angles. The proposed simulation model of the ion beam impact on SDO and the control algorithms can be used to formulate requirements to the control system for the SDO removal to the disposal orbit.

Acknowledgements This work was supported by the Ministry of Science and Higher Education of the Russian Federation. Agreement Number: 05.604.21.0211. Unique identifier of the project: RFMEFI60419X0211.

References

1. Bombardelli, C., Pelaez, J.: Sistema de modificacion de la posicion y actitud de cuerpos en orbita por medio de satelites guia. Patent number P201030354 (2010). PCT Patent Application PCT/ES2011/000011
2. Bombardelli, C., Merino, M., Ahedo, E., Pelaez, E., Urutxua, Y., Herrera, J., Iturri, A., Olimpio, A., Summerer, L., Petkow, D.: Active removal of space debris—ion beam shepherd for contactless debris removal. Ariadna Final Report, ESA Contract No. 4000101447/100/NL/CBi, P 992011 (2011)
3. Alpatov, A., Khoroshylov, S., Bombardelli, C.: Relative control of an ion beam shepherd satellite using the impulse compensation thruster. Acta Astronaut. **151**, 543–554 (2018)
4. Loginov, S., Usovik, I., Yakovlev, M., Obukhov, V., Popov, G., Svotina, V., Vilkov, Yu., Kirillov, V., Popov, V.: Contactless removal of space debris objects from the defending GEO region. Kosmonavtica I Raketostroenie **5**(98), 28–36 (2017) (in Russian)
5. Obukhov, V., Pokryshkin, A., Popov, G., Svotina, V.: Stability of a moving control of a service SC and a space debris object at impact on it by an ion beam. In: Razoumny, Yu.N., Graziani, F., Guerman, A.D., Contant J.-M. (eds.) Advances in the Astronautical Sciences DyCoSS'2017, vol. 161, pp. 665–675. Moscow, Russia (2017)
6. Nadiradze, A., Obukhov, V., Popov, G., Svotina, V., Pokryshkin, A.: Modeling of the force impact on a large-sized object of space debris by ion injection. In: Joint Conference of 30th ISTS, 34th IEPC and 6th NSAT, pp. 1–8. Hyogo-Kobe, Japan (2015)
7. Nadiradze, A., Obukhov, V., Pokryshkin, A., Popov, G., Svotina, V.: Modeling of the force and erosion action of ion beam on a large-sized object of space debris of technogenic nature. Izvestiya Academii Nauk, Energetika **2**, 146–157 (2016) (in Russian)
8. Cichocki, F., Merino, M., Ahedo, E., Smirnova, M., Mingo, A., Dobkevicius, M.: Electric propulsion subsystem optimization for "Ion BEAM Shepherd" missions. J. Propuls. Power (in print)
9. Balashov, V., Cherkasova, M., Kruglov, K., Kudriavtsev, A., Masherov, P., Mogulkin, A., Obukhov, V., Riaby, V., Svotina, V.: Radio frequency source of a weakly expanding wedge-shaped xenon ion beam for contactless removal of large-sized space debris objects. Rev. Sci. Instr. **88**(8), 083304.1–083304.5 (2017)

Chapter 11
Application of Low-Power Pulse Plasma Thrusters in Thrust Units of Small Spacecrafts

Aleksander V. Bogatiy, Grigory A. Dyakonov, Roman V. Elnikov, and Garri A. Popov

Abstract The chapter considers the current state of work on flight models of pulsed plasma propulsion systems. It is shown that the primary application area for propulsion systems based on an ablative pulsed plasma thruster is the station-keeping of small spacecraft with the power of supply system of up to 100 W and with an active lifetime in a range from 1 to 10 years in low Earth orbits with altitudes in a range from 400 to 700 km. It is also shown that ablative pulsed plasma thrusters can be efficiently used to solve the problems of accurate attitude control and angular stabilization of spacecraft.

11.1 Introduction

Currently, SpaceCraft (SC) with a mass of 10–1000 kg belonging to the class of Small SC (SSC). The development of new electronic and optical technologies allows us to fundamentally change the appearance and capabilities of SSC. Thus, at present SSC with a mass of up to 100 kg can often have a payload with characteristics comparable to those of a large spacecraft [1].

SSC can quite efficiently solve such urgent tasks as remote Earth sensing, navigation, mapping, communications, especially if they are combined into orbital systems, including two or more spacecraft with optoelectronic, radar, and other equipment that

A. V. Bogatiy (✉) · G. A. Dyakonov · R. V. Elnikov · G. A. Popov
Research Institute of Applied Mechanics and Electrodynamics of the Moscow Aviation, Institute (RIAME MAI), 5 Leningradskoye schosse, Moscow 125080, Russian Federation
e-mail: boga-alex@yandex.ru

G. A. Dyakonov
e-mail: grigory987@yandex.ru

R. V. Elnikov
e-mail: elnikov_rv@mail.ru

G. A. Popov
e-mail: riame@sokol.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_11

provide high resolution due to the summation of apertures of individual spacecraft equipment [2].

The operating conditions of most of such spacecraft require the maintenance and regular correction of their orbits, which makes it necessary to use small-sized propulsion systems that could operate efficiently in conditions of limited power consumption. The growing demands on the accuracy of keeping the parameters of SSC orbits, as well as, on the thruster lifetime and total pulse, necessitate the use of Electric Propulsion Systems (EPS) on such SSC. The current level of SSC power-weight ratio is about 1 W/kg. The limited mass of SSC and the limited power of their onboard power plants, as well as, the restrictions imposed on the cost of their development and operation, require the development of small, lightweight, and cheap electric propulsion systems with high efficiency in a range of power consumption of about 50–100 W and slightly higher.

This chapter is organized as follows. Section 11.2 reviews the types of EPS and their applications. Section 11.3 assesses the requirements for EPS for a low-orbit SSC. Section 11.4 discusses the possibility of solving the problem of SSC orientation using EPS. Section 11.5 compares APPT-based EPS with other types of EPS. Section 11.6 describes new development at the RIAME MAI of propulsion system based on APPT for SSC, its calculated technical characteristics are given. Section 11.7 concludes the chapter.

11.2 Types of Low-Power Electric Propulsion Systems

The main types of low-power Electric Propulsion Thrusters (EPT) classified by the mechanism of propellant acceleration are shown in Fig. 11.1.

In Electrothermal Thrusters (ETT), the energy of the outflowing gas is determined by its temperature upstream the nozzle. Thrusters of such class include the

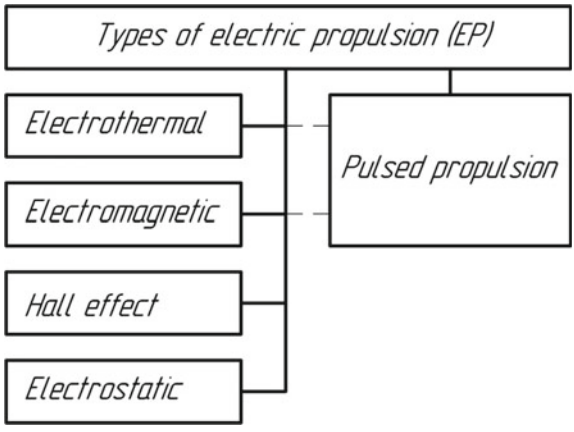


Fig. 11.1 Types of electric propulsion

thermo-catalytic, electrothermal, and arcjet thrusters. Hall thrusters (HT) with closed electron drift have two varieties: Stationary Plasma Thruster (SPT) and Thruster with Anode Layer (TAL) [3, 4]. Thrusters with electromagnetic acceleration are usually divided into two subclasses: magneto-plasma-dynamic thrusters with self-induced and applied magnetic field. Electrostatic thrusters include ion thrusters (IT) and colloid thrusters with various mechanisms of propellant ionization [5, 6]. Pulsed Plasma Thrusters (PPT) are usually considered as a separate class [7] due to the pronounced specificity of operating processes. It should be noted that in Ablative PPT (APPT) both electrothermal and electromagnetic plasma acceleration can take place. Often, the mixed mechanism of acceleration is implemented.

The main condition for the application of EPT of any type as a part of SSC electric propulsion system is the possibility for its operation in limited power consumption conditions. In this chapter, the power consumption of 100 … 150 W is assumed as the boundary value. This condition can be met by the following types of electric propulsion thrusters (only EPT that reached the stage of flight tests or purpose-oriented operation in space conditions are considered), i.e. in:

- Class of electrothermal thrusters includes the hydrazine-operated thermo-catalytic thrusters, electrothermal thrusters, and arcjet.
- Class of HT involves SPT and TAL.
- Class of electrostatic thrusters joints the ion thrusters and colloid thrusters.
- Special class of pulsed thrusters contains APPT, usually with Teflon (fluoroplast-4) as a propellant, in which the acceleration physics is either electrothermal (arc) or electromagnetic with a self magnetic field, or mixed.

In [8], an attempt was made to determine the preferred areas of application of various types of electric propulsion by analyzing the published data. Figure 11.2 shows the results of such analysis. If one knows the values of the necessary thrust F and total pulse J_Σ, N·s, obtained from the design calculation of the spacecraft

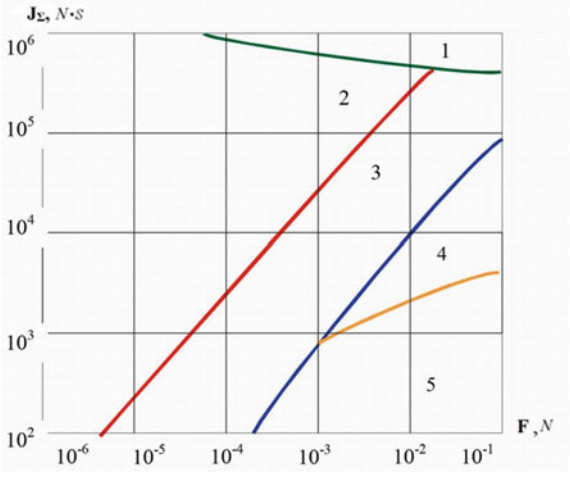


Fig. 11.2 Preferred EP application [8], where 1—IT, 2—IT and HT, 3—HT and APPT, 4—APPT and ETT, 5—ETT

and defined by SC purpose-oriented task, one can approximately determine the appropriate type of propulsion system using the diagram in Fig. 11.2.

The diagram in Fig. 11.2 shows that the choice of EPS type for SSC is not an easy task and it should be made at the stage of preliminary design taking into account the task to be performed by the spacecraft and the purpose of the propulsion system. Two typical tasks that can be performed by a low-power electric propulsion system (power consumption from 10 to 100 W and slightly higher) are considered hereinafter: low Earth orbit maintenance and attitude control for a spacecraft.

11.3 Low Earth Orbit Maintenance for Small Spacecraft

One of the typical tasks for low-power electric propulsion systems is to maintain a relatively low circular near-Earth orbit of the spacecraft. The possibility of solving this problem using APPT was studied in a number of papers.

The aerodynamic drag force F_a acting on a spacecraft moving in orbit at a velocity V is described as [9]:

$$F_a = 1/2 \cdot C_d \cdot \rho V^2 \cdot S_m, \tag{11.1}$$

where ρ is the density of the atmosphere gases (to a first approximation, except for its fluctuations in solar radiation, it depends only on the orbit altitude h above the Earth and is regulated by GOST 4401–81 for the International Standard Atmosphere (ISA)), C_d is the aerodynamic drag coefficient (for a free-molecular gas flow that occurs at densities corresponding to the upper atmosphere ($h > 200$ km), $C_d \approx 2.3$) [9], S_m is the midsection area of the spacecraft.

The spacecraft velocity, in the simplest case of a circular orbit with altitude h, is defined by Eq. 11.2 [9], where G is the gravitational constant, M is the mass of the Earth, R_E is the average radius of the Earth.

$$V^2 = G \cdot M/(R_E + h) \tag{11.2}$$

The characteristic velocity V_x necessary to maintain a conditional circular orbit with altitude h during time T is equal to:

$$V_x = F_a \cdot T/m, \tag{11.3}$$

where m is the spacecraft mass.

Figure 11.3 shows the calculated dependences of the aerodynamic drag force F_a averaged in accordance with ISA and characteristic velocity V_x necessary to maintain circular orbit of a conventional small satellite with mass $m = 100$ kg and midsection area of 1 m^2 in the orbit with altitude h for one year ($T \approx 3.16 \times 10^7$ s). When calculating, the aerodynamic drag coefficient was assumed as $C_d = 2.3$.

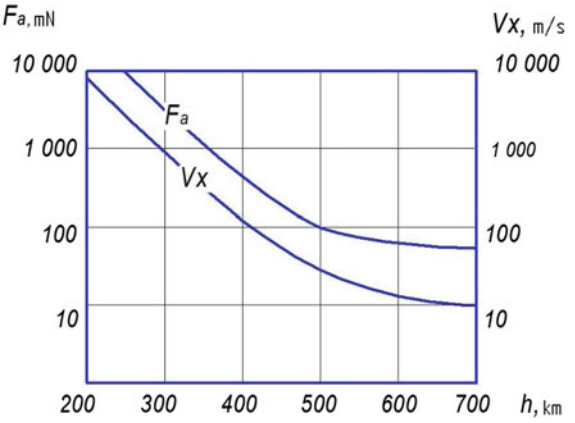


Fig. 11.3 The calculated dependences of the aerodynamic drag force F_a and of the characteristic velocity V_x necessary to maintain circular orbit of conventional SSC ($m =$ 100 kg, $S_m = 1$ m^2) on the orbit altitude h, for one year

The characteristic velocity is related to the parameters of the propulsion system by the Tsiolkovsky formula represented by Eq. 11.4, where J_{sp} is the specific impulse (the mass-averaged efflux velocity) of the propulsion system, m is the total mass of SSC (taking into account the mass of propellant), m_p is the propellant store.

$$V_x = J_{\text{sp}} \cdot \ln\left[m/(m - m_p)\right] \tag{11.4}$$

For electric propulsion systems, as a rule, $m_p \ll m$. Therefore, Eq. 11.4 can be replaced by a simpler ratio without compromising accuracy:

$$V_x = J_{\text{sp}} \cdot m_p/m, \tag{11.5}$$

or

$$V_x = J_\Sigma/m, \tag{11.6}$$

where $J_\Sigma = J_{\text{sp}} \cdot m_p = m \cdot V_x$ is the total pulse of the propulsion system.

Figure 11.4 shows the calculated dependences of the required total pulse J_Σ on the time of maintaining a low circular orbit of a conventional SSC weighing 100 kg and having a midsection area of 1 m^2 for various altitudes h of a low circular near-earth orbit.

It is known that the International Standard Atmosphere is not recommended for calculating orbits of artificial Earth satellites, since it does not take into account significant fluctuations in the density of the upper atmosphere depending on the time of day, season, and solar activity. However, such a simplified approach allows us to obtain estimates for the minimum thrust of the electric propulsion and total pulse of EPS required to maintain the orbit of a given altitude and is quite acceptable for assessing the possibility of using a propulsion system of this or that type.

More complicated mission analysis taking into account atmospheric fluctuations is presented in [9]. The minimum, maximum, and average estimates of the average

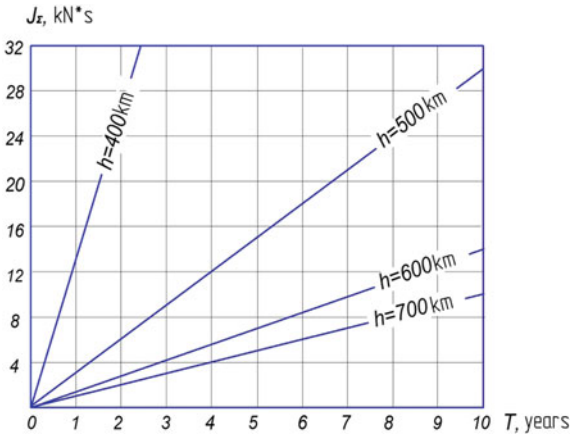


Fig. 11.4 The calculated dependences of the required total pulse of EPS on the given time of maintaining low circular orbit of conventional SSC ($m =$ 100 kg, $S_m = 1$ m^2) for different orbit altitudes h

annual total pulse of the electric propulsion system required to maintain circular orbits of various altitudes h were obtained. The calculation results are presented in Table 11.1 and are in good agreement with the approximate curves presented in Fig. 11.4.

We can see from the data presented in Fig. 11.4 and Table 11.1 that the necessary total pulse for keeping SSC orbit with an altitude in the range from 400 to 700 km for a period from 1 to 10 years is ranging from 1 to 30 kN s. If it is required to remove SC from its orbit after the end of its lifetime, the required total pulse nearly doubles. In this case, the averaged aerodynamic drag force of a spacecraft with a frontal area of 1 m^2 at the altitude of 400 km and above does not exceed 0.4 mN, which allows us to use various electric propulsion thrusters with a thrust of at least 1 mN for maintaining such orbits. The ratio of the averaged aerodynamic drag force to EPS thrust approximately equals the ratio of EPS operation time at each orbit pass to the orbit period—the relative propulsion time.

Table. 11.1 EPS total pulse for the year of flight necessary to maintain a circular orbit (SSC midsection area is 1.0 m^2), kN s [9]

Orbit height, km	Minimum estimate	Maximum estimate	Mean estimate
250	97.58	224.77	173.54
300	25.59	79.59	57.34
350	7.97	32.55	22.29
400	2.76	14.61	9.62
500	0.39	3.36	2.07
600	0.06	0.90	0.54
800	0.06	0.12	0.09

11.4 Attitude Control and Angular Stabilization

The second task that can be assigned to EPS of SSC is the attitude control and angular stabilization of the spacecraft. The reactive, flywheel, and gyro-force systems for the spacecraft attitude control are known, while there is a tendency to gradually abandon the use of reactive systems that require the consumption of propellant [10]. The peculiarity of the flywheel and gyro-force attitude control systems is that they require periodic dumping of the accumulated kinematic momentum, which requires an additional system for unloading the attitude control system, which, in turn, can be reactive, magnetic (using the Earth's magnetic field) and, less often, gravitational or aerodynamic. Currently, for low-orbit SSC, the most widely used are the flywheel and gyro-force attitude control systems in combination with a magnetic dumping system. Such attitude control systems do not require the consumption of propellant to control SSC motion relative to its center of mass. The problems associated with increased consumption of electric power and an additional mass of electromechanical attitude control systems have been successfully solved. An electromechanical attitude control system based on flywheels with magnetic unloading was applied even on such a lightweight scientific SSC as "Chibis-M" weighing 42 kg only [11]. Nevertheless, reactive attitude control systems comprising pulsed plasma thrusters are also characterized by low mass and extremely low propellant consumption. The first use of APPT for the system of solar panel attitude control took place at the end of 1964 on the spacecraft "Zond-2", developed by the RSC "Energia" [12].

When very precise SSC attitude control is required, for example, for remote Earth sensing satellites, the reactive attitude control systems based on the pulsed electric propulsion with a very small single thrust pulse are still beyond the competition. In particular, a reactive attitude control system with APPT was used on the remote-sensing satellite "EO-1" with an active lifetime of about 10 years [13]. In that case, the reactive attitude control system operated together with the electromechanical attitude control system, providing precise pointing of optical surveillance devices. The total pulse of a single EPS module required for that was just 0.46 kN s.

11.5 Available Family of Pulsed Plasma Thrusters and Their Rational Application Areas

Currently, a number of electric propulsion systems based APPT with discharge energy from 8 to 155 J are developed at the RIAME MAI, which is shown in Fig. 11.5 [7]. All propulsion systems of the APPT series are intended mainly for correcting and maintaining the orbit of low-orbit SSC. The most advanced of them are the following: APPT-45–2, APPT-155, and APPT-95 EPS. They passed the full range of ground experimental testing (in the case of APPT-95, with the exception of lifetime tests). APPT-45–2-based EPS designed for the scientific small spacecraft "MKA-FKI PN2" was launched into low Earth orbit in 2014. APPT-155-based EPS

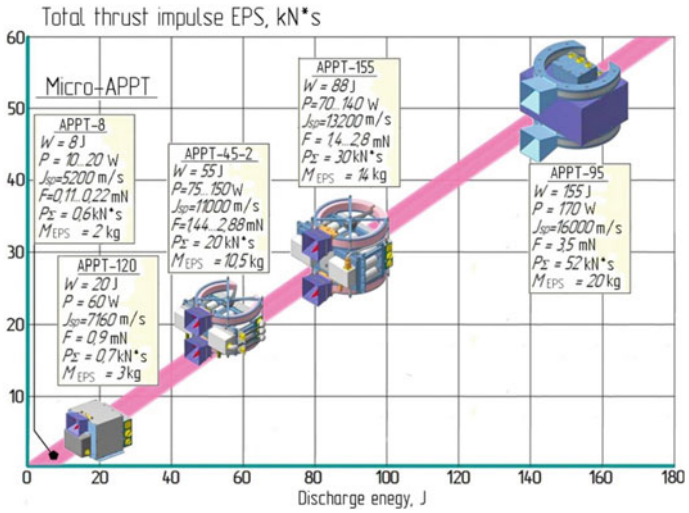


Fig. 11.5 Family of EPS based on APPT developed at the RIAME MAI (APPT-95 is being developed jointly with NIIEM) [7]

was designed for the small remote-sensing satellite "Soyuz-Sat-O" and differed from APPT-45–2 by the increased total pulse since its functions included not only maintaining the orbit, but also satellite deorbiting at the end of its active life. The most powerful of the presented EPS family is APPT-95-based, which is being developed jointly with the Research Institute of Electromechanics (NIIEM) for the scientific SSC "Ionosphere-M" [14].

Comparison of various types of electric propulsion in order to define the place of APPT among them is to be carried out for the whole electric propulsion system, which includes, in addition to the thruster itself, Propellant Storage and Feed System (PSFS) with full propellant store and Power Processing Unit (PPU). PPU, as a rule, makes up a significant (up to 50%) fraction of the total EPS mass. Currently, there is a tendency to reduce this fraction due to the transition to more advanced electronic components and digital control under the common protocol with SSC. Table 11.2 shows the characteristics of several types of EPS with a power consumption of about 100 W and a total mass of about 10 kg:

- Electrothermal thruster DUMIT was developed by the Production Association "Polet" [7] (prototype of the flight model). Thrusters of such type are widely used on satellites developed by VNIIEM and NIIEM [15].
- Radio-frequency ion thruster RIT-4 (Germany) [16, 17] (laboratory model).
- Stationary plasma thruster SPT-25 was developed by the Experimental Design Bureau "Fakel" [8] (laboratory model).
- Ablative pulsed plasma thrusters APPT-45–2 and APPT-250.

Table. 11.2 Comparative characteristics of the main EPS types with power consumption of about 100 W

EPS	DUMIT	RIT-4	SPT-25	APPT-45–2	APPT-250
Thruster type	ETT	RIT	SPT	APPT	APPT
Propellant	Ammonium	Xenon	Xenon	Teflon	Teflon
Power consumption, W	97	82	98	75 or 150	60 or 120
Average thrust, mN	30	2.5	4.7	1.45…2.9	1.2…2.4
Total EPS mass less propellant store, kg	10.0	10.0*	10.5*	8.7	6.9
Propellant store, kg	4.0	2.0*	2.0*	1.8	1.3
Total pulse, kN s	10	63.8*	16.2*	20	15.6
Thrust efficiency	0.62	0.49	0.22	0.11	0.12
Specific impulse, m/s	2500	31,900	8,100	11,000	12,000
Thrust cost, W/mN	3.2	32.8	20.8	52	50
Effective specific impulse, m/s	714	5,320*	1,300*	1,900	1,900

*Computed values

The latter one is an improved version of APPT-45–2 and has the same power consumption level. It differs in the use of new energy storage capacitors of the domestic company "NUKON" having higher specific energy and a digital control based on new electronic components. This allowed us to reduce the total mass of EPS from 10.5 kg down to 8.2 kg, in particular, the mass of two-channel PPU having dual redundancy was reduced from 1.5 kg down to 1.0 kg.

It should be noted that the thrusters RIT-4 and SPT-25 exist so far as laboratory models only, thus the unknown values of total pulse and total EPS mass are replaced by the values calculated based on the assumption of a propellant store of 2.0 kg and approximate specific masses of PSFS and PPU by analogy with the published data for SPT-50-based EPS [15]. A number of well-known propulsion systems that have passed flight tests as part of SSC, for example, those based on SPD-50 and RIT-10, are not included in Table 11.2, as they have other levels of power consumption, i.e. more than 200 W and more than 500 W, respectively.

As we can see from Table 11.2, the process of selecting an electric propulsion system for SSC is not straightforward. Different categories of thrusters have advantages and disadvantages, thus the choice should depend on the mission performed by the spacecraft. In RIAME MAI papers [18], a single criterion was proposed for comparing various types of electric propulsion systems—the effective specific impulse, which is numerically equal to the ratio of the total pulse of the propulsion system to its total mass taking into account the mass of the power processing unit and mass of the propellant storage and feed system with the full propellant store. According to this criterion, with an assumed propellant store of 2 kg, the ion propulsion systems are significantly superior to EPS of other types, thus their use as a part of SSC is advisable for tasks associated with high requirements for the total pulse.

At the same time, pulsed plasma thrusters have their own advantages. In APPT, the thrust and power consumption are controlled by changing the frequency of the pulses at constant discharge energy. As a result, their specific characteristics, such as specific impulse and thrust efficiency, are independent of power consumption. At the same time, for stationary electric propulsion, this dependence does not take place: with a decrease in power consumption, the specific characteristics of the thruster are decreasing, too. When the power of the electric propulsion system is below a certain threshold, the pulsed plasma thrusters will be superior to stationary ones in basic characteristics. Besides, the pulsed plasma thrusters have a number of additional advantages:

- Simplicity of design and electrical circuit and lack of expensive materials, which leads to a low cost of the electric propulsion system as a whole.
- Low weight and ease of control of PPU having one channel only for converting low onboard voltage to high (or two identical channels in the case of dual redundancy).
- Simplicity and reliability of the solid propellant storage and feed system, which does not have pipelines, valves, or other fittings.
- Cheap and non-deficient propellant, which is usually a fluoroplast-4 (Teflon).
- Constant operation readiness and extremely precise value of the thrust pulse, which is explained by the accuracy of defining the small magnitude of impulse bit.
- Monoblock design of the electric propulsion system, which does not require separate slots for PPU and PSFS to be provided for in SSC design.

The overall dimensions of the available domestic APPT with a power consumption of about 100 W make it possible to store enough propellant to provide a total pulse of up to 15–30 kN s. Based on this fact, it is possible to determine the predominant field of application of APPT-based electric propulsion system—the maintenance of small spacecraft with a power of supply system of up to 100 W and with an active lifetime in a range from 1 to 10 years in low Earth orbits with altitudes of 400 km to 700 km.

In addition, it is necessary to take into account the possibility of APPT using as a part of the precise attitude control systems of SSC for remote-sensing and other purposes. Table 11.3 shows the characteristics of APPT-based electric propulsion

Table. 11.3 Comparative characteristics of electric propulsion systems based on low-power APPT designed for SSC attitude control

EPS	EO-1	APPT-120
Power consumption, W	60	60
Average thrust, mN	0.86	0.90
Total EPS mass with propellant store, kg	4.05	3.0
Total pulse, kN s	0.46	0.7
Impulse bit, mN s	0.86	0.45
Specific impulse, m/s	10,400	7,160
Effective specific impulse, N s/kg	93	233

system designed for the mentioned purpose: EPS of "EO-1" satellite [13] and its analog, APPT-120 developed at the RIAME MAI [7]. EO-1 and APPT-120 belong to the same class of thrust and power consumption. Some difference in individual characteristics is explained by different schemes of the discharge channel. An extremely low impulse bit (of the order of 0.5 … 1.0 mN s) is interesting, which makes thrusters of this type the most promising for fine attitude control systems of SSC.

11.6 Immediate Opportunities for Further Development of Electric Propulsion with Ablative Pulse Plasma Thrusters

At present, APPT-based Corrective Electric Propulsion System (CEPS) of autonomous remote-sensing small spacecraft, which has high thrust and power, mass-and-size characteristics, is being developed at the RIAME MAI. APPT CEPS continuing the line of development of APPT-45–2 and APPT-250 refers to the same size of propulsion systems in terms of power consumption, thrust, and total pulse. One of the main characteristics of a pulsed plasma thruster, which determines its size and technical appearance, is the discharge energy (energy content) of a capacitor energy storage unit. It was shown in [18] that the most promising way to improve the mass-and-size characteristics of an electric propulsion system based on APPT is to make a capacitor energy storage unit less heavy by optimizing its energy content and using power capacitors with increased energy storage density. Optimization of the energy content of the capacitor bank can reduce the total mass of the electric propulsion system by 10 … 15%.

In the aforementioned paper, the structural diagram of the propulsion system based on APPT is analyzed and the estimative calculations of CEPS total mass depending on the discharge energy are performed for the following given total pulse J_Σ values: 10 kN s, 30 kN s, 50 kN s, and for the energy storage density of the capacitors ω_C = 28 J/kg, which corresponds to the lightest of the currently used capacitors of the MSR25 type produced by ICAR (Italy) and domestic pulse capacitors by Nukon. The calculation results are shown in Fig. 11.6.

We can see from the diagrams presented in Fig. 11.6 that the discharge energy of the available APPT-based CEPS models (the diagrams show the experimental points corresponding to APPT-155 and APPT-95) are significantly higher than the optimum one from the point of view of obtaining the minimum mass of EPS. The RIAME MAI experience in APPT development and testing shows that a decrease in discharge energy is accompanied by an increase in the thrust cost (C_T). Sometimes the C_T value is of great importance for a low-power SSC. The experimentally obtained dependence of the thrust cost C_T of flight models of APPT-based propulsion systems on the discharge energy is shown in Fig. 11.7.

Taking into account the above analysis of the influence of the energy content of the capacitor bank on the total mass of the electric propulsion system when developing

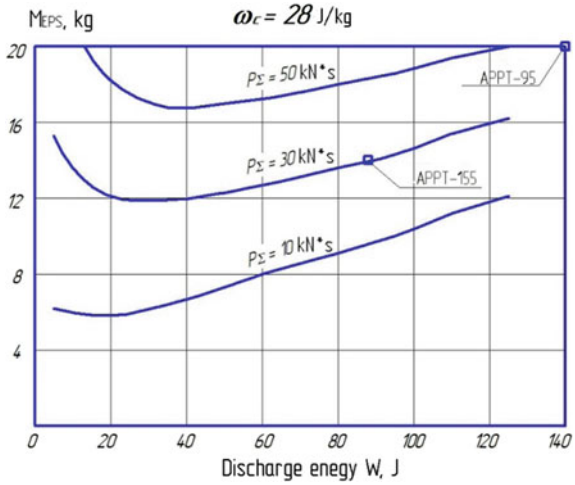


Fig. 11.6 Calculated dependence of APPT-based CEPS total mass on the discharge energy [18]

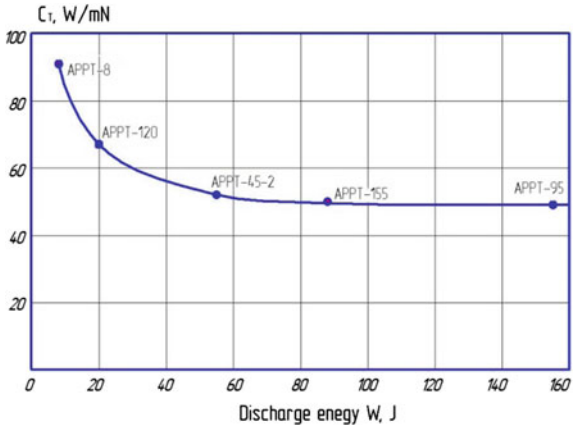


Fig. 11.7 APPT-based CEPS thrust cost as a function of discharge energy [18]

a new APPT-350 thruster, the discharge energy was reduced compared with APPT-155 and APPT-95 thrusters, down to 50 J, which allowed to reduce the total mass of the electric propulsion system with a propellant store down to 8 kg. The calculated characteristics of APPT-350 EPS are given in Table 11.4.

11.7 Conclusions

In this chapter, the current state of work on flight models of pulsed plasma propulsion systems is considered. It is shown that the primary application area for propulsion systems based on an ablative pulsed plasma thruster is the station-keeping of small

Table. 11.4 Design characteristics of APPT-350-based propulsion system

Discharge energy, J	50
Power consumption, W 1st thrust mode 2nd thrust mode	50 100
Specific impulse, m/s	10,000 at least
Average thrust, mN 1st thrust mode 2nd thrust mode	1.25 2.5
Total pulse, kN s	15
Mass of usable propellant, kg	1.3
Total EPS mass with propellant store, PPU including, kg	8.0

spacecraft with the power of supply system of up to 100 W and with active lifetime in range from 1 to 10 years in low Earth orbits with altitudes ranging from 400 to 700 km. For such spacecraft, the pulsed plasma thrusters are superior to stationary thrusters of various types in terms of the primary specific characteristics of electric propulsion systems and also have significant advantages in production and operation costs. Another field of EPS application is narrower, but is it the pulsed plasma thrusters are practically beyond the competition in terms of the accuracy of the thrust pulse produced—these are the systems for precise attitude control of small spacecraft.

Acknowledgements This work was carried out as a part of the implementation of the Federal Target Program "Research and Development in Priority Directions for the Development of the Russian Science and Technology Complex for 2014–2020" (Agreement No. 05.607.21.0308. Unique Agreement Identifier: RFMEFI60719X0308).

References

1. Grundmann, J., Meß, J., Biele, J., Seefeldt, P., Dachwald, B., Spietz, P., Grimm, C., Spröwitz, T., Lange, C., Ulamec, S.: Small spacecraft in small solar system body applications. In: EEE Aerospace Conference, pp. 1–20. Montana, United States (2017)
2. Schaffer, S., Chien, S., Branch, A., Hernandez, S.: Automatic orbit selection for a radio interferometric spacecraft constellation. J. Aerospace Inf. Syst. **15**(1), 1–13 (2018)
3. Kim, V., Gnizdor, R., Grdlichko, D., Zakharchenko, V., Korkunov, M., Merkurev, D., Popov, G., Shilov, E.: Development of an SPT-100VT stationary plasma thruster with increased thrust. Cosm. Res. **57**, 301–309 (2019)
4. Kaufman, H.: Theory of ion acceleration with closed electron drift. J. Spacecraft and Rockets **21**(6), 1–21 (1985)
5. Antropov, N.N., Akhmetzhanov, R.V., Bogatiy, A.V., Grishin, R.A., Kozhevnikov, V.V., Plokhikh, A.P., Popov, G.A., Khartov, S.A.: Experimental research of radio-frequency ion thruster. Therm. Eng. **63**(13), 957–963 (2016)
6. Groh, K.H., Loeb, H.W.: State-of-the-art of radio-frequency ion thrusters. J. Propul Power **7**(4), 573–579 (1989)

7. Zhang, Z., Ling, W.Y.L., Tang, H. Cao, J., Liu, X., Wang, N.: A review of the characterization and optimization of ablative pulsed plasma thrusters. Rev. Mod. Plasma Phys. **3**(1), 5 (2019)
8. Kulkov, V.M., Obukhov, V.A., Egorov, Y.G., Belik, A.A., Kraynov. A.M.: A comparative evaluation of the effectiveness of promising types of electric rocket thrusters as part of small spacecraft. Bull. Samara State Aerosp. Univ.**3**(34), 187–195. (2012). (In Russian)
9. Vorobev, A.L., Elnikov, R.V.: Analysis of the structure of families of locally optimum solutions to the problem of the interplanetary transfer of a spacecraft with a low—thrust engine. Cosm. Res. **56**(5), 365–372 (2018)
10. Mehrjardi, M.F., Sanusi, H., Ali, M.A., Abdullah, M.: Integrated attitude-drbit dynamics and control of spacecraft systems: State of the art and future trends. IEEE Aerosp. Electron. Syst. Mag. **33**, 60–71 (2018)
11. Ivanov, D.S., Ivlev, N.A., Karpenko, S.O., Ovchinnikov, M.Y., Roldugin, D.S., Tkachev, S.S.: The results of flight tests of the Chibis-M microsatellite orientation system. Space Res. **52**, 205–215 (2014)
12. Pets, L.A., Simonov, A.I., Khrabrov, V.A.: How to create the first electric propulsion. Earth Universe **6**, 57–60 (2005). (In Russian)
13. Zakrzwski, C., Benson, S., Sanneman, P., Hoskins, A.: On-orbit testing of the EO-1 pulsed plasma thruster. AIAA **2002–3973**, 1–11 (2002)
14. Makridenko, L.A., Volkov, S.N., Gorbunov, A.V., Kozhevnikov, V.A., Khodnenko, V.P.: Space complex "Ionosonde". In: Proceedings of VNIIEM 1703 Problems of Electromechanics, pp. 40–48 (2019). (In Russian)
15. Khodnenko V.P., Kolosova M.V.: Corrective propulsion systems for advanced spacecraft for remote sensing of the Earth. Actual problems of Russian cosmonautics. In: Proceedings of the XXXVII Academic Readings in Space, pp. 98–100 (2013). (In Russian)
16. Baruth, T., Thueringer, R., Klar, P.: Radiated emission simulation of a RIT 4. Joint Conference of 30th ISTS, 34th IEPC and 6th NSAT, pp. 1–9. Hyogo-Kobe, Japan (2015)
17. Bulit, A., Luna, J.P., Lotz, B., Feili, D., Leiter, H.J.: Experimental Investigations on the influence of the facility background pressure on the plume of the RIT-4 ion engine. In: 32nd International Electric Propulsion Conference, pp. 1–10. Wiesbaden, Germany (2011)
18. Bogatiy, A.V., Dyakonov, G.A., Maryashin, A.Y., Nechaev, I.L., Popov, G.A., Khalapyan, K.G.: Prospects for improving the weight and size characteristics of ablative pulsed plasma thrusters. In: Proceedings of VNIIEM Problems of Electromechanics ,vol. 133, issue no. (2), pp. 19–26 (2013). (In Russian)

Part III
Computational Solid Mechanics

Chapter 12
Multi-mode Model and Calculation Method for Fatigue Damage Development

Ilia S. Nikitin, Nikolay G. Burago, Alexander D. Nikitin, and Boris A. Stratula

Abstract A multi-mode kinetic model of damage development under cyclic loading is proposed to describe the process of fatigue failure. To determine the coefficients of the kinetic equation of damage, the well-known criterion of multiaxial fatigue failure is used. A procedure is proposed for calculating the kinetic equation coefficients for various fatigue failure modes of the LCF-HCF and VHCF. A numerical method for calculating crack-like zones up to macrofracture is developed. The model parameters are determined from the condition of matching the experimental and calculated fatigue curve for a specimen of a certain geometry at a given load amplitude and cycle asymmetry coefficient. Using the obtained values, the results of experiments on specimens of a different geometry and asymmetry coefficients were reproduced and the model and calculation algorithm performance were confirmed.

12.1 Introduction

Entire classes of criteria have been constructed that relate the number of cycles before the initiation of fatigue damage (microcracks) with the amplitudes and maximum values in the cycle (or average) that characterize the uniform stress-strain state of the working part of the specimen in a fatigue test.

I. S. Nikitin (✉) · N. G. Burago · A. D. Nikitin · B. A. Stratula
Institute for Computer Aided Design of the RAS, 19/18, Vtoraya Brestskaya ul., Moscow 123056, Russian Federation
e-mail: i_nikitin@list.ru

N. G. Burago
e-mail: buragong@yandex.ru

A. D. Nikitin
e-mail: nikitin_alex@bk.ru

B. A. Stratula
e-mail: stratula@matway.net

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_12

A large number of stress-based criteria are based on a direct generalization of the S-N Wöhler-type curves described by Basquin-type relations [1], and based upon the results of fatigue tests. The main criteria for multiaxial fatigue failure, taking into account the values of strain amplitudes (strain-based criteria), were proposed in [2–4]. These criteria are divided into two large groups. The first group includes criteria that use the amplitudes of the invariant characteristics of the stress state in the loading cycle such as octahedral stresses, principal stresses, etc. [5, 6], and the second group includes criteria that take into account the amplitudes of the tangent and/or normal stresses on the so-called critical plane [7–14]. As a rule, this plane is determined from the condition of the maximum amplitudes of the tangent, normal stresses, or a certain combination of them on the planes of various orientations. Reviews on this topic are given, say, in [15–19].

In order to study the development of fatigue damage zones, there are also two approaches. The first is based on the classical concepts of fracture mechanics and relates the conditions for the development of fatigue cracks depending on the amplitudes of stress intensity factors at the crack tip with the increase in the number of cycles. The basic equation was proposed by Paris and Erdogan [20], there are a large number of modifications of it [21–23]. The second approach uses representations of the theory of damage, dating back to [24, 25] and developed in [26–28]. As applied to the problems of cyclic loading and fatigue failure, it was used in [29–32].

We study the processes of fatigue damage zones development using the damage theory approach dating back to [24, 25]. In the application to the cyclic loading and fatigue failure problems, this approach was applied in [27, 28]. We propose a multimode model for the development of fatigue failure based on the evolutionary equation for the damage function. The model parameters are determined for various modes of fatigue failure: Low-Cycle Fatigue (LCF) and High-Cycle Fatigue (HCF), as well as, the regime of Very-High-Cycle Fatigue (VHCF), corresponding to high-frequency low-amplitude loading.

To distinguish the various modes of fatigue failure, we use the multimode amplitude fatigue curve diagram shown in Fig. 12.1. Up to a value of $N \sim 10^3$, the regime of re-static loading is realized with an amplitude that differs little from the static strength limit σ_B. Further, the left part of the bimodal fatigue curve (Wöhler curve) describes LCF-HCF modes up to $N \sim 10^7$ and amplitude values of the order of the fatigue limit σ_u. Then begins the zone of change of fracture mechanisms and a further drop in fatigue strength, starting from $N \sim 10^8$ to a new fatigue limit value $\tilde{\sigma}_u$ in accordance with the right branch of the bimodal S-N fatigue curve. This branch describes VHCF mode [33].

It should be noted that at present, the idea of an explicit division of the classic Wöhler branch into two parts (in fact, LCF and HCF) exists. The boundary of this transition region is determined not by the value of N, but by the value of the loading amplitude equal to the yield strength of the material σ_T [34] since this changes the physical mechanism of fatigue failure. In addition, the boundary of the repeated-static range $N \sim 10^3$ is rather arbitrary. It is also specified in [34] depending on the strength and plastic characteristics of the material. However, in this chapter, we keep

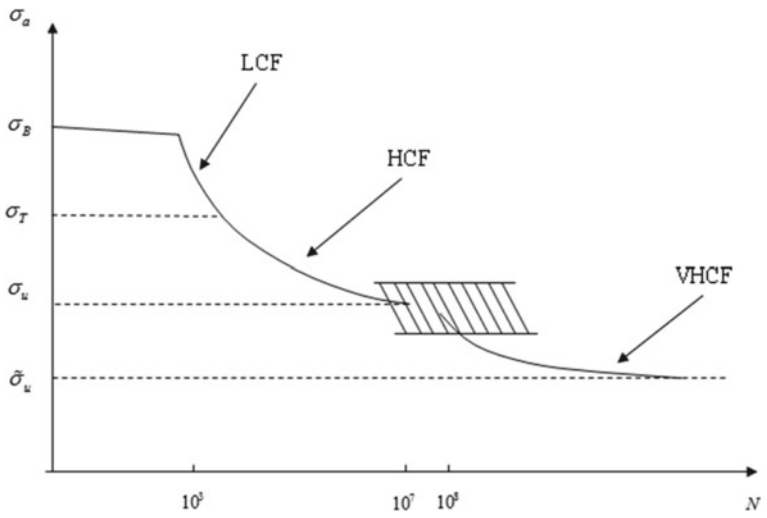


Fig. 12.1 Bimodal fatigue curve

the suggestion of the proposed model of damage development based on the scheme described above.

In order to match the model with the well-known criteria for multiaxial fatigue failure, a stress-based criterion has been selected that describes the fatigue failure associated with the normal crack microcracks development. This is a modification of the Smith–Watson–Topper (SWT) criterion [4] described in [35], in which the amplitudes of maximum tensile stresses play a decisive role in the development of fatigue damage.

This chapter is organized as follows: Sect. 12.2 presents kinetic equations for two fatigue modes and a condition to switch between them. Algorithm for fatigue damage development calculation is presented in Sect. 12.3. Section 12.4 is dedicated to calculation results both in LCF-HCF and VHCF modes. Section 12.5 concludes the chapter.

12.2 Kinetic Equation for Damage

In this work, two different modes of fatigue loading are studied: classic LCF-HCF mode and VHCF mode in Sects. 12.2.1 and 12.2.2, respectively. Also, an algorithm to choose between them is presented in Sect. 12.2.3.

12.2.1 LCF-HCF Mode

The criterion of multiaxial fatigue failure in LCF-HCF mode with the development of normal crack microcracks [35] (stress-based SWT) corresponding to the left branch of the bimodal fatigue curve (Fig. 12.1) is

$$\sqrt{\langle\sigma_{1_{\max}}\rangle\Delta\sigma_1/2} = \sigma_u + \sigma_L N^{-\beta_{\text{LH}}}, \tag{12.1}$$

where σ_1 is the largest principal stress, $\Delta\sigma_1$ is the range of the largest principal stress per cycle, $\Delta\sigma_1/2$ is the amplitude. From the condition of repeated-static fracture up to values of $N \sim 10^3$ by the method [19] it is possible to obtain the value $\sigma_L = 10^{3\beta_{\text{LH}}}(\sigma_B - \sigma_u)$. According to the chosen criterion only tensile stresses lead to failure, thus it includes the value $\langle\sigma_{1_{\max}}\rangle = \sigma_{1_{\max}} H(\sigma_{1_{\max}})$. In these formulas σ_B is the static tensile strength of the material, σ_u is the classic fatigue limit of the material during a reverse cycle (asymmetry coefficient of the cycle $R = -1$), β_{LH} is power index of the left branch of the bimodal fatigue curve.

From the fatigue fracture criterion we obtain the number of cycles before fracture at a uniform stressed state:

$$N_{\text{LH}} = 10^3[(\sigma_B - \sigma_u)/\langle\sigma_{\text{LH}} - \sigma_u\rangle]^{1/\beta_{\text{LH}}}, \sigma_{\text{LH}} = \sqrt{\langle\sigma_{1_{\max}}\rangle\Delta\sigma_1/2}. \tag{12.2}$$

In order to describe the process of fatigue damage development in LCF-HCF mode, a damage function $0 \le \psi(N) \le 1$ is introduced, which describes the process of gradual cyclic material failure. When $\psi = 1$, a material particle is considered completely destroyed. Its Lame modules become equal to zero. The damage function ψ as a function on the number of loading cycles for LCF-HCF mode is described by the kinetic equation:

$$\mathrm{d}\psi \big/ \mathrm{d}N = B_E \psi^{\gamma}/(1 - \psi^{\alpha}),$$

where α and $0 < \psi < 1$ are the model parameters that determine the rate of fatigue damage development. The choice of the denominator in this two-parameter equation, which sets the infinitely large growth rate of the zone of complete failure at $\psi \to 1$, is determined by the known experimental data on the kinetic growth curves of fatigue cracks, which have a vertical asymptote and reflects the fact of their explosive, uncontrolled growth at the last stage of macro fracture.

An equation for damage of a similar type was previously considered in [29], its numerous parameters and coefficients were determined indirectly from the results of uniaxial fatigue tests. In our case, the coefficient B_{LH} is determined by the procedure that is clearly associated with the selected criterion for multiaxial fatigue failure of one type or another. It has the following form. The number of cycles to complete failure N_{LH} at $\psi = 1$ is defined from the equation for damage for a uniform stress state provided by Eq. 12.3.

$$\left[\psi^{1-\gamma}/(1-\gamma) - \psi^{(1+\alpha-\gamma)}/(1+\alpha-\gamma)\right]\Big|_0^1 = B_{\mathrm{LH}}\, N|_0^{N_E}$$
$$N_{\mathrm{LH}} = \alpha/(1+\alpha-\gamma)/(1-\gamma)/B_{\mathrm{LH}} \tag{12.3}$$

By equating the values N_{LH} from the fracture criterion (Eq. 12.2) and from the solution of the equation for damage (Eq. 12.3), we obtain the expression for the coefficient B_{LH}:

$$B_{\mathrm{LH}} = 10^{-3}[\langle\sigma_{\mathrm{LH}} - \sigma_u\rangle/(\sigma_B - \sigma_u)]^{1/\beta_{\mathrm{LH}}}\alpha/(1+\alpha-\gamma)/(1-\gamma),$$

where the value σ_{LH} is determined by the selected mechanism of fatigue failure and the corresponding multiaxial criterion (Eq. 12.1).

12.2.2 VHCF Mode

The criterion of multiaxial fatigue failure in VHCF mode corresponding to the right branch of the bimodal fatigue curve (generalized stress-based SWT) (Fig. 12.1) has the form:

$$\sqrt{\langle\sigma_{1_{\max}}\rangle\Delta\sigma_1/2} = \tilde{\sigma}_u + \sigma_V N^{-\beta_{\mathrm{VH}}}.$$

Here, from the similarity condition of the control points for the left and right branches of the bimodal fatigue curve [33], we can obtain the formula:

$$\sigma_V = 10^{8\beta_{\mathrm{VH}}}(\sigma_u - \tilde{\sigma}_u).$$

From the criterion of fatigue failure, we obtain the number of cycles to failure in a uniform stress state:

$$N_{\mathrm{VH}} = 10^8\left[(\sigma_u - \tilde{\sigma}_u)/\langle\sigma_{\mathrm{VH}} - \tilde{\sigma}_u\rangle\right]^{1/\beta_{VH}},\ \sigma_{\mathrm{VH}} = \sigma_{\mathrm{LH}} = \sqrt{\langle\sigma_{1_{\max}}\rangle\Delta\sigma_1/2},$$

where $\tilde{\sigma}_u$ is the fatigue limit of the material during the reverse cycle for VHCF mode, β_{VH} is the power-law index of the right branch of the bimodal fatigue curve.

12.2.3 Condition for Switching the Modes of Accumulation of Fatigue Damage

The transition point from the left branch of the fatigue curve to the right branch, at which the mechanism of fatigue fracture changes, is slightly above the fatigue limit σ_u (Fig. 12.1) and is determined by the value $\sigma_* = \sigma_u + \Delta\sigma$. To ensure continuous

conjugation of the left and right branches of the fatigue curve, it is necessary to fulfill a condition $N_{\mathrm{LH}} = N_{\mathrm{VH}}$ that is equivalent to the equation for the quantity $\Delta\sigma$:

$$10^3[(\sigma_B - \sigma_u)/\langle\Delta\sigma\rangle]^{1/\beta_{\mathrm{LH}}} = 10^8\left[(\sigma_u - \tilde{\sigma}_u)\Big/\langle\sigma_u + \Delta\sigma - \tilde{\sigma}_u\rangle\right]^{1/\beta_{\mathrm{VH}}}$$

or

$$\Delta\sigma = 10^{-5\beta_{\mathrm{LH}}}(\sigma_B - \sigma_u)\left[1 + \Delta\sigma/(\sigma_u - \tilde{\sigma}_u)\right]^{\beta_{\mathrm{LH}}/\beta_{\mathrm{VH}}}.$$

Given the actual smallness of the correction term in square brackets, one can set the correction value $\Delta\sigma$ by an approximate formula:

$$\Delta\sigma = 10^{-5\beta_{\mathrm{LH}}}(\sigma_B - \sigma_u).$$

The corresponding approximate value $N_* = N_{\mathrm{LH}}(\sigma_*)$ is determined by

$$N_* = 10^3[(\sigma_B - \sigma_u)/\Delta\sigma]^{1/\beta_{\mathrm{LH}}} \approx 10^8.$$

Given the updated estimates obtained for the transition point from one branch of the fatigue curve to another, we obtain the final formulas for the ranges and coefficients of the kinetic equations of damage.

For LCF-HCF mode when $\sigma_u + \Delta\sigma_u < \sigma_{\mathrm{LH}} < \sigma_B$ and $\Delta\sigma = 10^{-5\beta_{\mathrm{LH}}}(\sigma_B - \sigma_u)$, we obtain:

$$B_{\mathrm{LH}} = 10^{-3}[\langle\sigma_{\mathrm{LH}} - \sigma_u\rangle/(\sigma_B - \sigma_u)]^{1/\beta_{\mathrm{LH}}}\alpha/(1 + \alpha - \gamma)/(1 - \gamma),$$

$$\sigma_{LH} = \sqrt{\langle\sigma_{1_{\max}}\rangle\Delta\sigma_1/2}.$$

For VHCF mode when $\tilde{\sigma}_u < \sigma_{VH} \le \sigma_u + \Delta\sigma_u$, we have:

$$B_{\mathrm{VH}} = 10^{-8}\left[\langle\sigma_{\mathrm{VH}} - \tilde{\sigma}_u\rangle\Big/(\sigma_u - \tilde{\sigma}_u)\right]^{1/\beta_{VH}}\alpha\Big/(1 + \alpha - \gamma)/(1 - \gamma),$$

$$\sigma_{\mathrm{VH}} = \sqrt{\langle\sigma_{1_{\max}}\rangle\Delta\sigma_1/2}.$$

When $\sigma_{\mathrm{VH}} \le \tilde{\sigma}_u$, fatigue failure doesn't occur; when $\sigma_{\mathrm{LH}} \ge \sigma_B$, it happens instantly.

12.3 Fatigue Damage Development Calculation Algorithm

Section 12.3 presents the approach to implement fatigue damage and calculate one's development. Ansys software was used to calculate the stress state within a loading

cycle of a deformable specimen supplemented by a code to calculate the damage equation and changes of elasticity modulus.

To integrate the equation $\mathrm{d}\psi \Big/ \mathrm{d}N = B\psi^{\gamma}/(1-\psi^{\alpha})$, where B either B_{LH} or B_{VH}, the damage function approximation was applied at the k-node of the computational grid for given discrete values ψ_k^n at moments N^n and sought ψ_k^{n+1} at moments N^{n+1}. To calculate the damage equation, the value $\alpha = 1-\gamma$ was chosen for which by analytic integration an explicit expression for $\psi_k^{n+1}(\psi_k^n, \Delta N^n)$ can be obtained:

$$\left[\psi^{1-\gamma}/(1-\gamma) - \psi^{2(1-\gamma)}/2/(1-\gamma)\right]\Big|_{\psi_k^n}^{\psi_k^{n+1}} = B\,N|_{N_n}^{N_{n+1}}.$$

With the denotations $(\psi_k^{n+1})^{1-\gamma} = x$, $q = 2(1-\gamma)B\Delta N^n + (\psi_k^n)^{1-\gamma} - 2(\psi_k^n)^{2(1-\gamma)}$ and $\Delta N^n = N^{n+1} - N^n$ the equation transforms to $x^2 - 2x + q = 0$ and its valid root $x = 1 - \sqrt{1-q} < 1$. The analytical expression for the increment of damage on the increment of the number of cycles ΔN^n is derived from:

$$\psi_k^{n+1} = \left(1 - \sqrt{1 - \left[2(1-\gamma)B\Delta N^n + (\psi_k^n)^{1-\gamma} - 2(\psi_k^n)^{2(1-\gamma)}\right]}\right)^{1/(1-\gamma)}.$$

Increment value ΔN^n is defined as follows. Based on the stress state calculation data, the coefficient $B = B_{\mathrm{LH}}, B_{\mathrm{VH}}$ is calculated for each node. After that, for each node, the following values are calculated by

$$\Delta\tilde{N}_k^n = \left[\psi^{1-\gamma}/(1-\gamma) - \psi^{2(1-\gamma)}/2/(1-\gamma)\right]\Big|_{\psi_k^n}^{1}/B$$

corresponding to the number of cycles, at which in the node k from its current level of damage and equivalent stress complete destruction will be achieved (damage is equal to 1). If the damage level in the considered node is less than the threshold ψ_0 (threshold $\psi_0 = 0.95$ is selected), then the value for this node $\Delta\tilde{N}_k^n$ is multiplied by a factor of 0.5. Otherwise, it is multiplied by a factor of 1. Thus, the step of incrementing the number of cycles for a given node is $\Delta N_k^n = 0.5(1 + H(\psi_k^n - 0.95))\Delta\tilde{N}_k^n$. Of all the ΔN_k^n values, the smallest one is selected. The increment of the number of loading cycles for the calculation of the entire specimen is $\Delta N^n = \min\limits_k \Delta N_k^n$. For each node based on its current level of damage and equivalent voltage, a new level of damage is found taking into account the calculated increment ΔN^n.

All elements are sorted out, for each of them the most damaged node is searched and according to its damage the mechanical properties of the element are adjusted:

$$\lambda(\psi_k^n) = \lambda_0(1 - \kappa\psi_k^n),\ \mu(\psi_k^n) = \mu_0(1 - \kappa\psi_k^n).$$

Those elements that belong to nodes with damage $\psi = 1$ are removed from the calculation area and form a localized zone (crack-like) of completely destroyed material. The calculation ends when the boundaries of a completely damaged region exit to the specimen surface (macro destruction) or the evolution of this region stops.

12.4 Calculation Results

Calculation of S-N curves and fatigue cracks propagation performed both in LCF-HCF and VHCF modes are presented in Sect. 12.4.

To determine the parameters of the proposed model and verify its performance, one of the fatigue experiments described in [29] was performed numerically. From the condition of matching the experimental and calculated fatigue curve for a specimen of certain geometry for a given loading amplitude and cycle asymmetry, the numerical coefficients were found. Using the obtained values, the experimental results on specimens of a different geometry and asymmetry coefficients were reproduced, and calculation algorithm operability was confirmed.

Hereinafter, the numerical results for LCH-HCF mode and VHCF mode are discussed in Sects. 12.4.1 and 12.4.2, respectively.

12.4.1 Results for LCH-HCF Mode

Initial tests were conducted on a plate 100 × 25 × 1.57 mm in size with 1.56 mm diameter through a hole in the center. Ratification tests were conducted on a V-notched specimen that has 15 mm width w/o a notch, thickness of 1.7 mm, a notch depth of 1.32 mm, a V-notch angle of 60 degrees, and a notch radius of 0.675 mm. The cyclic loading of the upper and lower boundaries of the specimen with an amplitude of 0.096 mm with the development of damage zones up to macroscopic destruction was simulated and matched with the results from [29]. In the center of the plate, there is a through-hole with diameter of 1.56 mm. Plate material is titanium alloy with strength and fatigue parameters $\sigma_B = 1135$ MPa, $\sigma_u = 30$ MPa, $\beta_{\text{LH}} = 0.31$. Elasticity modulus of intact alloy are $\lambda_0 = 77$ GPa, $\mu_0 = 44$ GPa. Figures 12.2 and 12.4 show the lines of the effective stress level σ_{LH} for the specimen with a hole (Fig. 12.2) and for the specimen with a notch (Fig. 12.4) in two states: before the fatigue quasi-crack initiation and at the moment when it has passed approximately halfway to macro-destruction.

In Figs. 12.3 and 12.5, the results of real and computational experiments on constructing fatigue curves for specimens with a hole and a side notch are presented. Both real and calculated points represent the moment of crack initiation. The curves in the figures approximate the experimental points. The calculations presented in Fig. 12.3b almost exactly fit the approximation curve for the values of the model parameters $\gamma = 0.1$ and $k = 0.5$. Utilizing these parameters, the fatigue curves are presented in Fig. 12.3a (specimen with a hole, $R = -1$) and in Fig. 12.5 (notched specimen, $R = -0.5$ and $R = 0.1$). In Fig. 12.3b, the relative error equals 0 for the calibration series. The average relative errors in Figs. 12.3a, 12.5a, b are 1%, 7%, and 6%, respectively. The obtained satisfactory quality reproduction of real fatigue experiments indicates the efficiency and prospects of the model and calculation algorithm. The considered model represents the development of the damage model in the

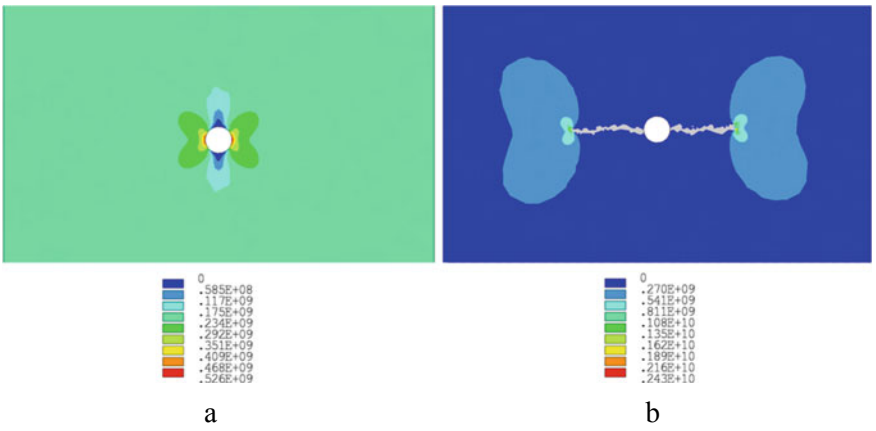


Fig. 12.2 Ti-alloy specimen with a hole at $R = -1$: **a** emergence of a "quasi-crack", **b** growth of a "quasi-crack"

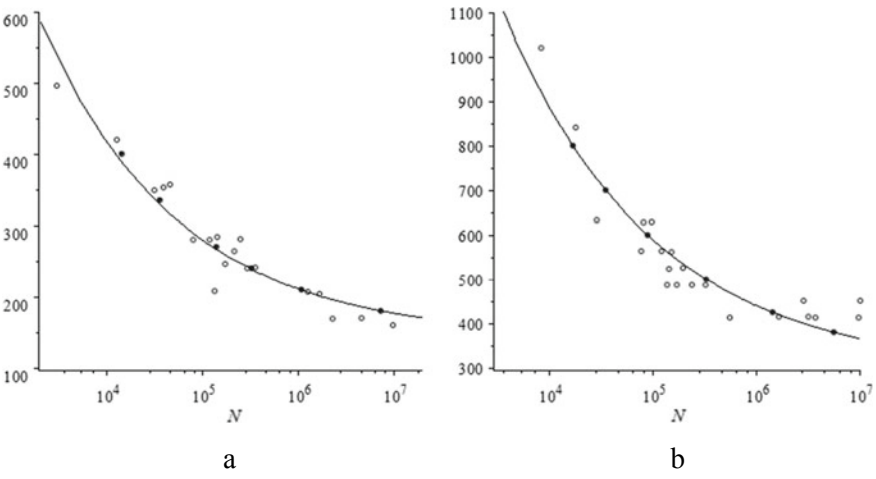


Fig. 12.3 Fatigue curves $\sigma_{\max}(N)$ for Ti-alloy specimen with a hole, where ∘ means real test points, • means calculating points: **a** $R = -1$, **b** $R = 0.54$

case of cyclic loads presented in [36] for the description of damages during dynamic loading.

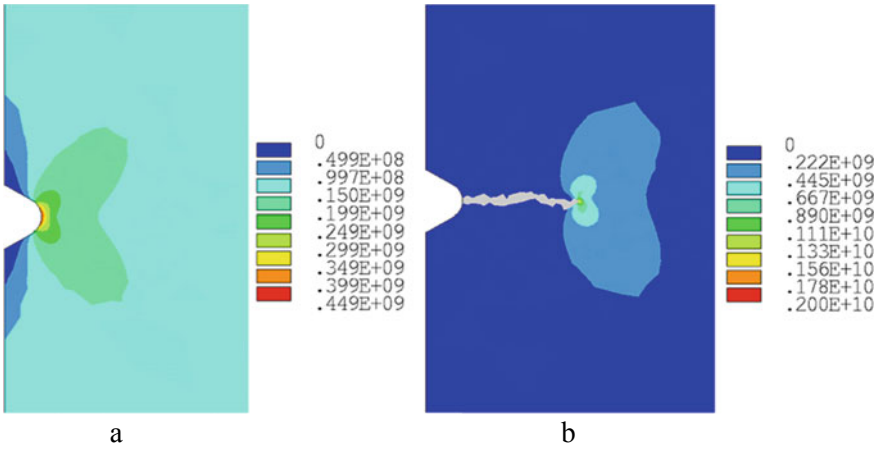


Fig. 12.4 V-notched Ti-alloy specimen at $R = -0.5$: **a** emergence of a "quasi-crack", **b** growth of a "quasi-crack"

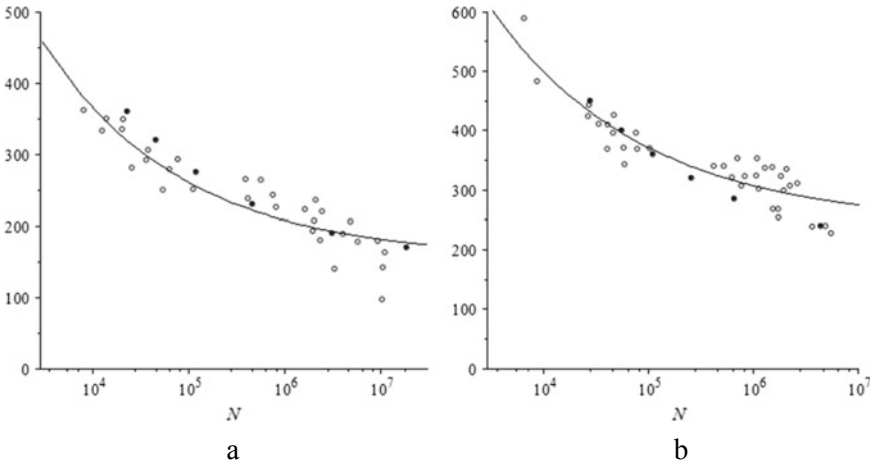


Fig. 12.5 Fatigue curves $\sigma_{\max}(N)$ for V-notched Ti-alloy specimen, where ∘ means real test points, • means calculating points: **a** $R = -0.5$, **b** $R = 0.1$

12.4.2 Results for VHCF Mode

In order to numerically investigate the development of crack-like regions of fatigue failure in UHMW mode, the cyclic loading of a specimen made of AS7G06-T6 aluminum alloy with reduced displacement amplitude of 0.1 mm was calculated. The corresponding experimental results are taken from the [37]. The mechanical properties of the Al-alloy: density $\rho = 2680$ kg/m^3, $E = 68$ GPa, tensile strength

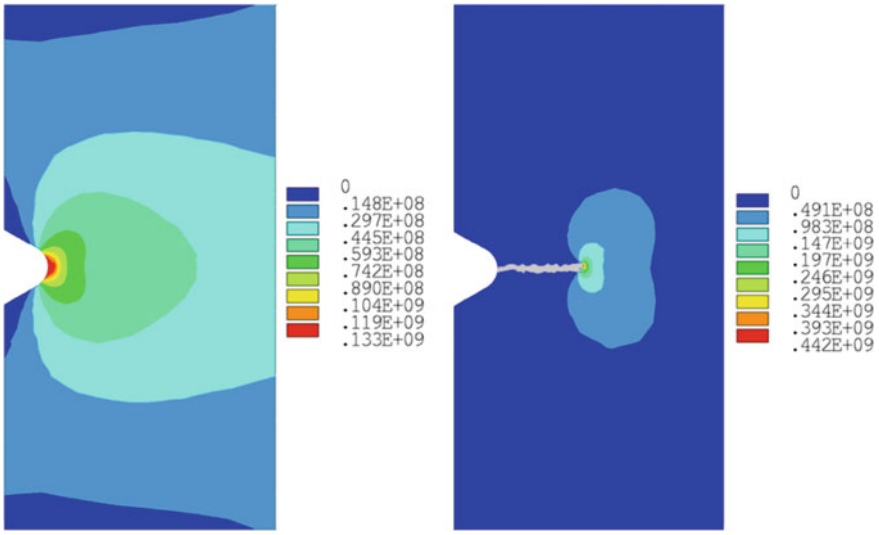


Fig. 12.6 V-notched Al-alloy specimen at $R = -1$: **a** emergence of a "quasi-crack", **b** growth of a "quasi-crack"

$\sigma_B = 288$ MPa, HCF fatigue limit $\sigma_u = 130$ MPa, VHCF fatigue limit $\tilde{\sigma}_u = 60$ MPa, $\beta_{\text{VH}} = 0.3$.

In the series of tests, the bi-curved notched specimen shape was used. In the waist, it has a quasi-flat shape of 6.18 mm width and 3 mm thick. The notch was 1 mm depth with the tip curvature radius of 0.5 mm and the angle of cleavage of 60 degrees.

Figures 12.6 and 12.7 show the calculation results for VHCF mode. In Fig. 12.7 the results of real and computational experiments on constructing fatigue curves for specimens with a side notch are presented. The curves in the figures approximate the experimental points for $R = -1$ (Fig. 12.7a) and $R = 0.01$ (Fig. 12.7b).

In Fig. 12.7, slight differences are observed between the calculated and experimental points. This can be explained as follows. The exponential exponent β_{VH} of the fatigue curve for aluminium weakly depends on the cycle asymmetry coefficient R [23, 37], but in the accepted calculating scheme with SWT criterion, this exponent is considered constant. Figure 12.6 shows the lines of the effective stress level σ_{VH} for the specimen with a notch in two states: before the fatigue quasi-crack initiation and at the moment when it has passed approximately halfway to macro-destruction.

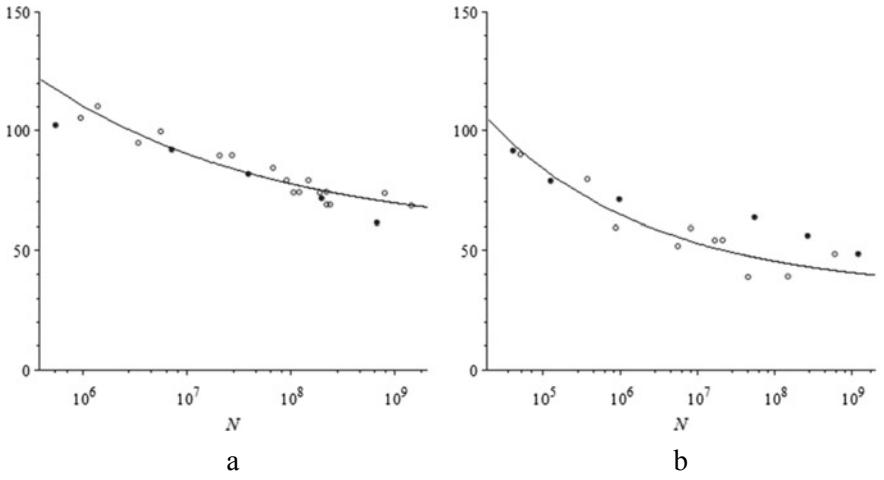


Fig. 12.7 VHCF fatigue curves $\sigma_{\max}(N)$ for V-notched Al-alloy specimen, where ∘ means real test points, • means calculating points: **a** $R = -1$, **b** $R = 0.01$

12.5 Conclusions

A multi-mode kinetic model of cyclic loading damage development is proposed to describe the fatigue fracture process development. To determine the coefficients of the kinetic equation of damage, the well-known criterion of multiaxial fatigue failure SWT based on the mechanism associated with the development of microcracks of normal detachment is used.

A procedure has been proposed for calculating the kinetic equation coefficients for various fatigue failure modes of the LCF-HCF and VHCF. A numerical method for calculating crack-like zones up to macrofracture is proposed. The model parameters are determined from the condition of matching the experimental and calculated fatigue curve for a specimen of a certain geometry at a given load amplitude and cycle asymmetry coefficient. Using the obtained values, the results of experiments on specimens of a different geometry and asymmetry coefficients were reproduced and the model and calculation algorithm performance were confirmed.

Acknowledgements The study was carried out with a grant from the Russian Science Foundation (project No. 19-19-00705).

References

1. Basquin, O.H.: The exponential law of endurance tests. Proc. Am. Soc. Test. Mater. **10**, 625–630 (1910)

2. Brown, M., Miller, K.J.: A theory for fatigue under multiaxial stress-strain conditions. Inst. Mech. Eng. **187**, 745–756 (1973)
3. Fatemi, A., Socie, D.F.: A critical plane approach to multiaxial damage including out-of-phase loading. Fatigue Fract. Eng. Mater. Struct. **11**(3), 149–156 (1988)
4. Smith, R.N., Watson, P., Topper, T.H.: A stress-strain parameter for the fatigue of metals. J. Mater. **5**(4), 767–778 (1970)
5. Sines, G.: Behaviour of metals under complex static and alternating stresses. In: Sines, G., Waisman, J.L. (eds.) Metal Fatigue, pp. 145–169. McGraw-Hill, New York (1959)
6. Crossland, B.: Effect of large hydrostatic pressures on torsional fatigue strength of an alloy steel. In: Proceedings of International Conference on Fatigue of Metals, pp. 138–149. London (1956)
7. Findley, W.: A theory for the effect of mean stress on fatigue of metals under combined torsion and axial load or bending. J. Eng. Ind. 301–306 (1959)
8. Morel, F.: A critical plane approach for life prediction of high cycle fatigue under multiaxial variable amplitude loading. Int. J. Fatigue **22**(2), 101–119 (2000)
9. Matake, T.: An explanation on fatigue limit under combined stress. Bull. JSME **20**, 257–263 (1977)
10. McDiarmid, D.L.: A shear stress based critical-plane criterion of multiaxial fatigue failure for design and life prediction. Fatigue Fract. Eng. Mater. Struct. **17**, 1475–1484 (1999)
11. Papadopoulos, I.V.: Long life fatigue under multiaxial loading. Int. J. Fatigue **23**, 839–849 (2001)
12. Carpinteri, A., Spagnoli, A., Vantadori, S.: Multiaxial assessment using a simplified critical plane based criterion. Int. J. Fatigue **33**, 969–976 (2011)
13. Susmel, L., Taylor, D.: A critical distance/plane method to estimate finite life of notched components under variable amplitude uniaxial/multiaxial fatigue loading. Int. J. Fatigue **38**, 7–24 (2012)
14. Suman, S., Kallmeyer, A., Smith, J.: Development of a multiaxial fatigue damage parameter and life prediction methodology for non-proportional loading. Frattura ed Integrità Strutturale **38**, 224–230 (2016)
15. Meggiolaro, M.A., Miranda, A.C., de Castro, J.: Comparison among fatigue life prediction methods and stress-strain models under multiaxial loading. In: Proceedings of 19th International Congress of Mechanical Engineering, Brasilia, DF, pp. 1–10 (2007)
16. Wang, Y.-Y., Yao, W.-X.: Evaluation and comparison of several multiaxial fatigue criteria. Int. J. Fatigue **26**, 17–25 (2004)
17. Karolczuk, A., Macha, E.: A review of critical plane orientations in multiaxial fatigue failure criteria of metallic materials. Int. J. Fatigue **134**, 267–304 (2016)
18. Karolczuk, A., Papuga, J., Palin-Luc, T.: Progress in fatigue life calculation by implementing life-dependent material parameters in multiaxial fatigue criteria. Int. J. Fatigue **134** (2020)
19. Bourago, N.G., Zhuravlev, A.B., Nikitin, I.S.: Models of multiaxial fatigue fracture and service life estimation of structural elements. Mech. Solids **46**(6), 828–838 (2011)
20. Paris, P.C., Erdogan, F.A.: Critical analysis of crack propagation laws. J. Basic Eng. **85**, 528–533 (1963)
21. Forman, R.G., Kearney, V.E., Engle, R.M.: Numerical analysis of crack propagation in a cyclic-loaded structure. Trans. ASME J Basic Eng. **89**(3), 459–464 (1967)
22. Collins, J.A.: Failure of Materials in Mechanical Design: Analysis, Prediction. Prevention. Wiley, NY (1993)
23. Bathias, C., Paris, C.P.: Gigacycle Fatigue in Mechanical Practice. Dekker, NY (2004)
24. Kachanov, L.M.: On the time of destruction under creep conditions. Izv. AN SSSR OTN 8, 26–31 (in Russian) (1958)
25. Rabotnov, J.N.: On the mechanism of long-term destruction. Voprosi prochnosti materialov i konstrukcij. AN SSSR OTN, 5–7 (in Russian) (1959)
26. Murakami, S.: Continuum Damage Mechanics. A Continuum Mechanics Approach to the Analysis of Damage and Fracture. Springer, Dordrecht (2012)

27. Altenbach, H., Skrzypek, J.J. (eds.): Creep and Damage in Materials and Structures. Springer Vienna, Vienna (1999)
28. Lemaitre, J., Chaboche, J.L.: Mechanics of solid materials. Cambridge University Press, Cambridge (1994)
29. Marmi, A.K., Habraken, A.M., Duchene, L.: Multiaxial fatigue damage modeling at macro scale of Ti6Al4V alloy. Int. J. Fatigue **31**, 2031–2040 (2009)
30. Zhi, Y.H., Wagner, D., Bathias, C., Chaboche, J.L.: Cumulative fatigue damage in low cycle fatigue and gigacycle fatigue for low carbon–manganese steel. Int. J. Fatigue **33**, 115–121 (2011)
31. Fincato, R., Tsutsumi, S.: Coupled damage-viscoplasticity model for metals under cyclic loading conditions. Procedia Struct. Int. **18**, 75–85 (2019)
32. Chaboche, J.L.: Time-independent constitutive theories for cyclic plasticity. Int. J. Plast **2**(2), 149–188 (1986)
33. Burago, N.G., Nikitin, I.S.: Multiaxial fatigue criteria and durability of titanium compressor disks in low- and giga- cycle fatigue modes. In: Mathematical Modeling and Optimization of Complex Structures, pp. 117–130. Springer, Heidelberg (2016)
34. Shanyavskiy, A.A., Soldatenkov, A.P.: The fatigue limit of metals as a characteristic of the multimodal fatigue life distribution for structural materials. Procedia Struct. Int. **23**, 63–68 (2019)
35. Gates, N., Fatemi, A.: Multiaxial variable amplitude fatigue life analysis including notch effects. Int. J. Fatigue **91**, 337–351 (2016)
36. Burago, N.G., Nikitin, I.S., Nikitin, A.D., Stratula, B.A.: Algorithms for calculation damage processes. Frattura ed Integrità Strutturale **49**, 212–224 (2019)
37. Perez-Mora, R.: Study of the fatigue strength in the gigacycle regime of metallic alloys used in the aeronautics and off-shore industries. Thèse de doctorat en Mécanique. https://www.theses.fr/2010ENAM0027# (2010)

Chapter 13
Elastic Wave Propagation Modeling During Exploratory Drilling on Artificial Ice Island

Igor B. Petrov, Maksim V. Muratov, and Fedor I. Sergeev

Abstract This chapter is devoted to numerical modeling of elastic impacts on artificial ice islands arising as a result of drill impacts while drilling from the island, earthquakes, and pressure of structures located on the island, as well as collisions of the ice island with drifting ice layers. To solve this problem numerically, we use the grid-characteristic method with interpolation on regular rectangular and parallelepipedal meshes and unstructured triangular and tetrahedral ones. The grid-characteristic method most accurately describes the dynamic processes in exploration seismology problems, since it takes into account the nature of wave phenomena. The approach used makes it possible to construct correct computational algorithms at the boundaries and contact boundaries of the integrational domain. In the work, the process of propagation of elastic waves in the considered geological environment studies simulates the distribution of stresses and also studies the stability of the ice island to destruction using the Mises criterion.

13.1 Introduction

Despite the active development of alternative energy technologies, oil and natural gas remain the main sources of energy throughout the world. At the same time, many traditional deposits, the technologies of exploration and development of which are well developed, are largely depleted. In recent years, various unconventional (hard-to-extract) deposits have been discovered, the work with which requires a development of science and technology in such areas as geomechanics, the theory of seismic waves,

I. B. Petrov · M. V. Muratov (✉) · F. I. Sergeev
Moscow Institute of Physics and Technology (National Research University), 9, Institutsky per, Dolgoprudny, Moscow 141701, Russian Federation
e-mail: max.muratov@gmail.com

I. B. Petrov
e-mail: petrov@mipt.ru

F. I. Sergeev
e-mail: sergeev.fi@phystech.edu

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_13

and so on. These facilities include deposits on the Arctic shelf. Most of hydrocarbon deposits are concentrated in the Arctic zone.

Artificial ice islands are used for mining of oil and gas in the Arctic. They represent a cheap and environmentally friendly alternative to conventional drilling platforms, making them well suited for exploratory drilling in offshore areas. Often this is a single way to produce the explorative drilling in shelf of northern sees, where due to severe ice conditions there is no possibility to deliver the usual platform. Such approach has already been successfully realized in Canada [1]. An actual problem for the safety of structures and personnel on the surface of the ice island, as noted in [2], is its destruction due to drilling and seismic activity. Wave processes resulting from drilling and earthquakes also affect the response to exploration seismology. Because of the limited possibility of conducting experimental studies in realistic conditions, the direction of numerical simulation is promising.

In this chapter, we consider a numerical simulation of the propagation of elastic waves in an ice island during exploration seismology and seismic activity. In modern computational software, the finite element method is used to study the stability of structures like ice islands [3, 4]. For seismic wave propagation modeling, researchers usually use the finite difference method [5], method of spectral elements [6], discontinuous Galerkin method [7, 8], and grid-characteristic method [9–12]. This work was made using the grid-characteristic method with interpolation on regular rectangular grids (in 2D case) and parallelepiped (in 3D case), as well as the unstructured triangular (in 2D case) and tetrahedral (in 3D case) meshes. This method is actively used also for seismic problems, for example, in [13]. It was chosen, since it allows one to set the correct boundary and contact conditions.

In Sect. 13.2, the considered problem formulations are described in detail. Section 13.3 is devoted to the mathematical model used and the numerical method. Section 13.4 presents the results of mathematical modeling of problems in the above formulations. Section 13.5 concludes the chapter.

13.2 Problem Formulation

The wave propagation simulation area is an ice island with length of 300 m, height of 10 m, surrounded by 8 m deep seawater, and resting on solid ground (Fig. 13.1). Ice, water, and soil are considered homogeneous. Elastic characteristics (longitudinal and transverse velocity of sound propagation and density of media) are given in Table 13.1.

The thickness of the seafloor is 10 m, and the thickness of the sedimentary rock is 600 m. A gas reservoir is located under the sedimentary rock, which is simulated by the boundary condition of the free boundary to simplify the task.

The following types of loads on the ice island are modeled:

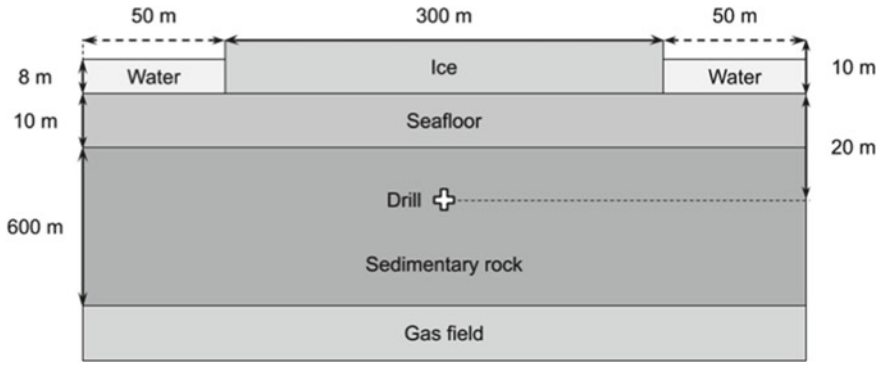


Fig. 13.1 Scheme of simulation area

Table 13.1 Elastic characteristics of described layers

Medium	*P*-wave speed (m/s)	*S*-wave speed (m/s)	Density (kg/m^3)
Ice	3940	2493	917
Water	1500	–	1025
Seafloor	1806	316	2000
Sedimentary rock	2250	1000	2000

- Drill impact. When drilling from ice island, the drill passes to a depth of 20 m, where it creates a point impact on the soil (Fig. 13.1). The work simulates the propagation of elastic waves generated by such an action.
- The impact of earthquakes. A plane wave as an earthquake wave is simulated from the depth. The stability of the ice island to its effects is studied.
- Exposure to static load (Fig. 13.2). On the island, there is a structure measuring 5 m × 5 m and weighing 5 tons. The loads on the ice island of this structure are calculated, as well as the limit values of the load parameters at which the island begins to collapse.

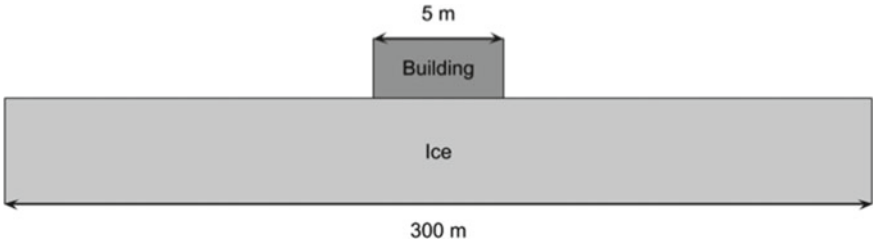


Fig. 13.2 Scheme of building location on ice island

- Side impact on an ice island drifting ice floes. A strike from the side of the island is modeled.

The results are recorded in the form of two-dimensional patterns of the distribution of elastic wave velocities in the medium, the components of the stress tensor, as well as the Mises stresses in the plane of the drilling well. Then the data obtained can be used for a detailed analysis of the results and their visual display in the form of two-dimensional visualizations.

13.3 Mathematical Model and Numerical Method

In this section, we consider a methodology of the problem solution. In Sect. 13.3.1, the mathematical model is represented. Section 13.3.2 is devoted to numerical method, while Sect. 13.3.3 is about the boundary and contact boundary conditions used in problem. The criterion of ice island destruction is described in Sect. 13.3.4, while Sect. 13.3.5 is devoted to the drill model used in problem with drill strikes.

13.3.1 Linear Elastic Medium Model

A complete system of equations of state of a continuous linear elastic medium and an acoustic field is solved [12].

$$\rho \frac{\partial V_i}{\partial t} = \frac{\partial T_{ji}}{\partial x_j}, \frac{\partial T_{ij}}{\partial t} = \lambda \left(\sum_k \frac{\partial V_k}{\partial x_k} \right) I_{ij} + \mu \left(\frac{\partial V_i}{\partial x_j} + \frac{\partial V_j}{\partial x_i} \right),$$

where V_i is the velocity components, T_{ji} is the components of tension tensor, ρ is the density of medium, λ and μ are the Lame coefficients, and I_{ij} is the component of unit tensor. Using vector of variables $\vec{u} = \left\{ V_x, V_y, V_z, T_{xx}, T_{yy}, T_{zz}, T_{xy}, T_{xz}, T_{yz} \right\}$, the equation system we can represent as

$$\frac{\partial \vec{u}}{\partial t} + \sum_{i=1,2,3} A_i \frac{\partial \vec{u}}{\partial \xi_i} = 0.$$

13.3.2 Grid-Characteristic Method

A numerical solution is found using the grid-characteristic method [9–14]. We carry out a coordinatewise splitting, and by changing variables, we reduce the system to a system of independent scalar transport equations in Riemann invariants:

$$\frac{\partial \vec{w}}{\partial t} + \Omega_i \frac{\partial \vec{w}}{\partial \xi_i'} = 0, i = 1, 2, 3.$$

For each transfer equation, all nodes of the computational mesh are bypassed, and characteristics are omitted for each node. From the time layer n, the corresponding component of the vector $\vec{w}$ is transferred to the time layer $n + 1$ by the formula:

$$w_k^{n+1}\left(\xi_i'\right) = w_k^n\left(\xi_i' - \omega_k \tau\right),$$

where τ is the time step.

After all the values are transferred, there is a reverse transition to the vector of the desired values of $\vec{u}$.

Interpolation on unstructured and regular grids is considered. Values at each point are found using values at grid reference points $\vec{w}\left(\vec{r}_{ijkl}\right)$ and weights of these points $p_{ijkl}(\vec{r})$ as

$$\vec{w}(\vec{r}) = \sum_{i,j,k,l} \vec{p}_{ijkl}(\vec{r})\vec{w}\left(\vec{r}_{ijkl}\right).$$

The grid-characteristic method allows the most correct algorithms to be applied at the boundaries and contact boundaries of the integration region [9, 10].

13.3.3 Boundary and Contact Boundary Conditions

The boundary condition can be written in general form as

$$\boldsymbol{D}\vec{u}(\xi_1, \xi_2, \xi_3, t + \tau) = \vec{d},$$

where $\boldsymbol{D}$ is some 9×3 matrix, $\vec{d}$ is a vector, and $\vec{u}(\xi_1, \xi_2, \xi_3, t + \tau)$ are the values of the desired velocity values and components of the stress tensor at the boundary point at the next time step.

At the boundaries of the integration region, the following boundary conditions were used:

1. At the side boundaries, absorbing (non-reflecting) boundary conditions are used:

$$v^n_{k-2} = v^n_{k-1} = v^n_k, T^n_{k-2} = T^n_{k-1} = T^n_k,$$

where index k corresponds to the boundary node of the grid, and k–1 и k–2 are its neighbors on one axis.

2. At the boundaries of the medium with air, the free boundary condition applies:

$$\boldsymbol{T}\vec{n} = 0.$$

3. The boundary condition of constant pressure is set at the contact of the building and the ice island and is written as follows:

$$p = T_{yy} = P_0, T_{xx} = T_{xy} = 0,$$

where P_0 is a constant pressure of building on ice surface.

On the contact boundaries between the layers, the following contact conditions are used.

1. The contact condition of complete adhesion is placed between the layers of solid media. Physically, it means the possibility of unhindered propagation of elastic waves. Mathematically, the condition of complete adhesion is written as follows:

$$\vec{v}_a = \vec{v}_b, \quad \vec{f}_a = -\vec{f}_b,$$

where $\vec{v}$ are the velocities of contact points, $\vec{f}$ is the force acting to the contact boundary, and a and b are the contact points of first and second contact layers.
2. The contact condition for free sliding is placed between the ice island and the ground. In contrast to the case of contact of two solid layers, when the condition of complete adhesion is applied, the ice and the bottom layer can move relative to each other. This phenomenon is known in practice, for example, glaciers are "slipping" from the surfaces of mountains. Thus, the use of a special contact condition is required:

$$\vec{v}_a \cdot \vec{n} = \vec{v}_b \cdot \vec{n}, \quad \vec{f}^a_n = -\vec{f}^b_n, \quad \vec{f}^a_\tau = \vec{f}^b_\tau = 0.$$

The same contact condition was used on contact between solid ground and water.

13.3.4 The Destruction Criterion

The chapter considers potential destruction at all points of the integration domain. Within the framework of the linear elastic medium model, it is possible to determine the possibility of fracture at a specific point. The destruction process itself is not considered. To determine the points of destruction, the Mises criterion is used [15]:

$$T_{\text{mises}} = \frac{1}{\sqrt{2}}\sqrt{(T_{xx} - p)^2 + \left(T_{yy} - p\right)^2 + (T_{zz} - p)^2 + 2T_{xy}^2 + 2T_{xz}^2 + 2T_{yz}^2} > y_s,$$

where $p = \left(T_{xx} + T_{yy} + T_{zz}\right)/3$ is the average stress, and y_s is the shear stress limit, which value for ice is equal to 1 MPa.

13.3.5 The Drill Model

Modern drills have a rather complex device, often combining percussion and rotational mechanisms. In this chapter, we are primarily interested in the wave pattern that occurs when drilling in a specific geological model. Therefore, for simplicity, we represent the impact of the drill as a point source with a Ricker pulse of 30 Hz and an amplitude coefficient 10^{12}. It can be written with formula:

$$R(t) = A\left(1 - 2\pi^2 f^2 t^2\right) exp\left(-\pi^2 f^2 t^2\right),$$

where A is the coefficient of compression, and f is the frequency.

Ricker pulse is represented in Fig. 13.3 for the case of frequency 0.4 Hz and amplitude coefficient 1.

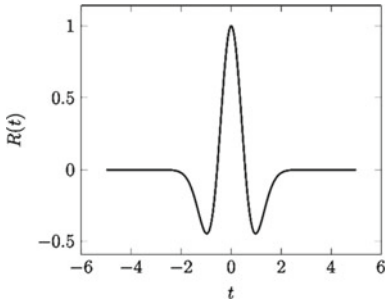


Fig. 13.3 Ricker pulse

13.4 The Results of Modeling

This section is devoted to the results of mathematical modeling. All the above statements are considered: the modeling of drill strikes, modeling earthquakes impacts on ice island, modeling of static loads on ice island, and modeling of collision of ice island with ice layer in Sects. 13.4.1 and 13.4.4, respectively.

13.4.1 The Modeling of Drill Strikes

The drill was set by a point source at a depth of 20 m with a Ricker wave pulse. We studied the process of propagation of waves from a drill in the medium, as well as the distribution of stresses in the ice island and its resistance to destruction. Figure 13.4 shows two-dimensional velocity fields in the medium under study at mentioned time instants. To study the resistance to fracture, the Mises stress distribution was constructed. Figure 13.5 shows a distribution of Mises stresses at mentioned time instants.

It was empirically determined that the island begins to collapse at the amplitude of the wave impulse caused by the impact of the drill 10^{12} Pa. This value is too large and cannot be in real problem.

The wave propagation from the drill can be divided into three segments. The first segment is the initial movement of an almost spherical wave from a source with active penetration into the ice island. The second segment is the movement of the wave deep into the solid rock and the reverse motion after reflection from the gas-bearing layer.

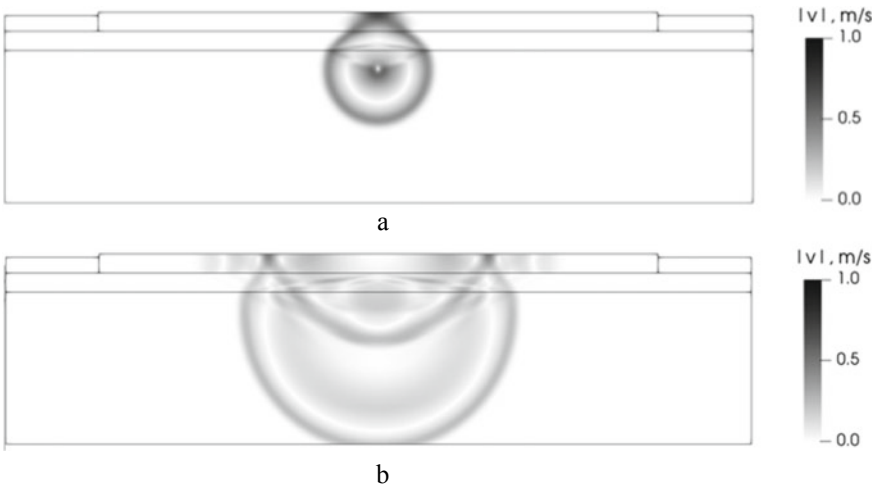


Fig. 13.4 Velocity fields from drill strike at time instants: **a** 0.0125 s, **b** 0.0325 s

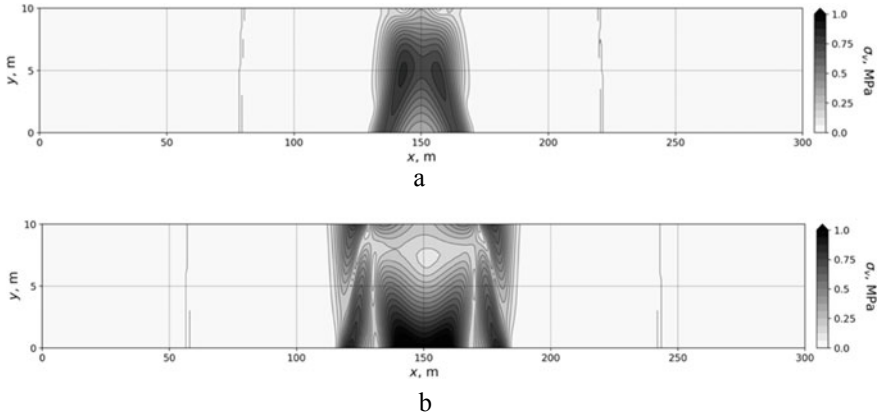


Fig. 13.5 Mises stresses distribution from drill strike at time instants: **a** 0.0125 s, **b** 0.0325 s

The third segment is the movement of the wave up the bottom layer with subsequent penetration into the island and reflection from the free surface of the ice.

We can see that the ice island plays the role of a kind of resonator. The free surface of the ice is completely reflective, while the boundary between the ice and the near-bottom layer partially reflects the elastic waves. Since the height of the island is small compared to its length, the vertically propagating wave experiences many reflections in the time it takes for a horizontally propagating wave to reach the edge of the island. This means that, possibly, with some special choice of external periodic disturbance, the ice island is able to accumulate elastic waves. Accumulation, of course, will occur until the beginning of the destruction of ice. Such a resonant phenomenon, if it exists, poses a significant danger to work on such an ice platform.

If the wave propagating from the introduced point source has a large amplitude, then it can destroy the ice. As can be seen from Fig. 13.5, such fractures will likely be located directly above the drilling point. This is facilitated by the interference of waves entering the island from below with waves reflected from the free surface of the ice.

13.4.2 The Modeling of Earthquake Impact on Ice Island

The earthquake was modeled as Ricker plane wave spreading from the depth to the surface. In Fig. 13.6, we can see the distribution of pressure (Fig. 13.6a) and Mises stresses (Fig. 13.6b) in the instant when the earthquake wave reaches the ice island.

Analyzing the Mises stresses distribution one can determine the points where the ice island is going to destruct.

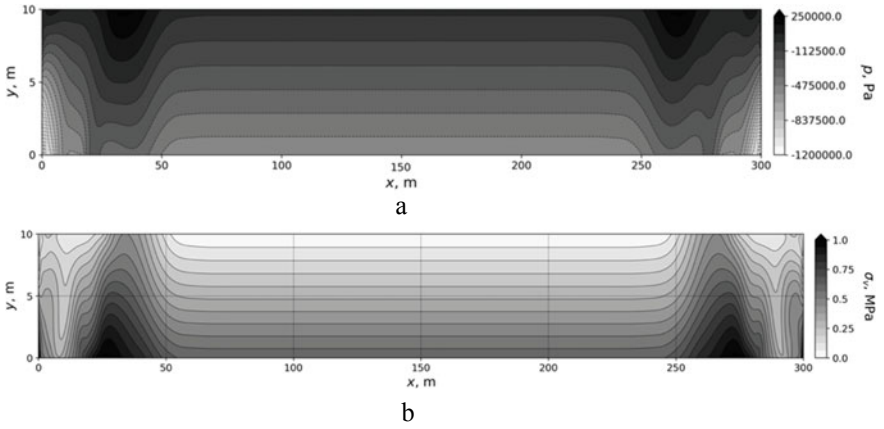


Fig. 13.6 Distribution under earthquake of: **a** pressure, **b** the Mises stresses

13.4.3 The Modeling of Static Loads on Ice Island

This task is relevant in the study of the ice island. A static load from a rig and other structures located on the island is constantly on the ice island. It is important to understand how much the ice island is resistant to such loads. We determine the stress distribution on the island.

To solve the problem of the distribution of static load, we use the establishment method [16]. We will consider the stress distribution in the ice island to be steady if the modulus of the propagation velocity of elastic waves appears to be 20–30 times lower at the final instant of time compared with the velocity at the initial instant.

We assume that the building located on the surface of the ice island has a mass of 100 tons and a base of 5 × 5 m, thereby producing pressure of 4 kPa on ice (Fig. 13.2). After completing 200 thousand steps, the velocity modulus decreased by approximately 33 times from 2.4×10^{-4} to 7.3×10^{-6} m/s. Thus, the resulting stress distribution can be considered steady.

Since only the pressure of the building on the horizontal surface of the ice is specified by external conditions, it is natural that T_{yy} will be the majority of the total pressure p and, accordingly, the Mises stress Tv. In Fig. 13.7, the distributions of pressure and the Mises stresses are represented. Maximal Mises stress in problem is 2.2 kPa. This value is less than shear stress limit of ice 1 MPa. Thus, at realistic values of the static load, the ice will not break.

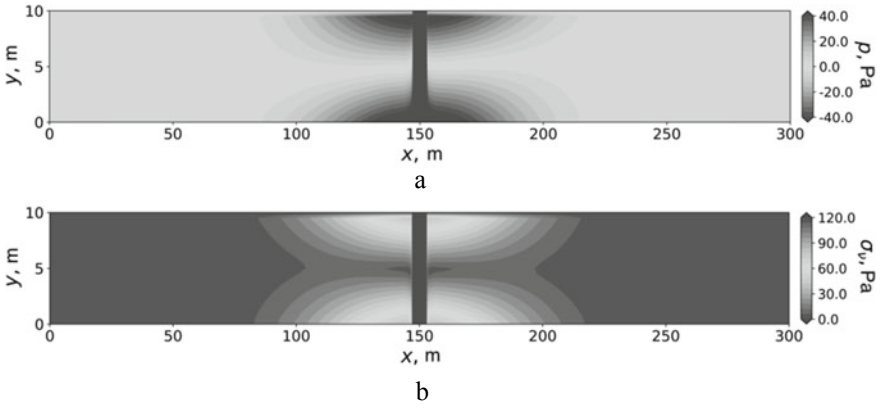


Fig. 13.7 Distribution under static load impact of: **a** pressure, **b** the Mises stresses

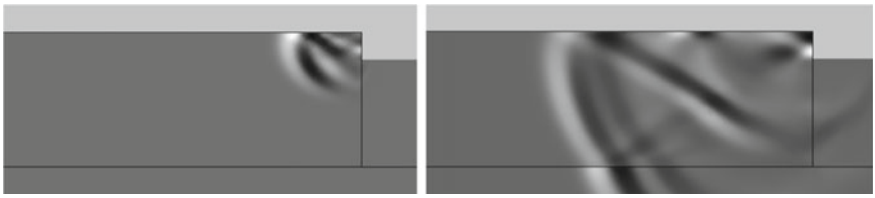


Fig. 13.8 Stress propagation by side impact: **a**—the initial moment of collision, **b**—the stated elastic wavefront propagating through ice island

13.4.4 The Modeling of Collision of Ice Island with Ice Layer

The collision of ice island with drifting ice layer modeled as impact of external force on side of ice island. The process of stress propagation is represented in Fig. 13.8.

The maximum stress values are reached the collision site and propagate in the form of shear vibrations along the upper boundary of the ice island. In the case of strong collisions in these areas, the destruction is possible.

13.5 Conclusions

The approach based on grid-characteristic method allows to model of ice island stability problems. We simulated the drill impact, impact of earthquakes, exposure to static load, and side impact on an ice island drifting ice floes. The approach helps to estimate influence of different impacts on ice island and determine their critical

value which leads to island destruction. These results can be used in research and explorational works on ice island.

Acknowledgements This work was carried out with the financial support of the Russian Science Foundation, project no. 19-11-00023.

References

1. Crawford, A., Crocker, G., Mueller, D., Desjardins, L., Saper, R., Carrieres, T.: The canadian ice island drift, deterioration and detection (CI2D3) database. J. Glaciology **64**(245), 517–521 (2018)
2. Petrov, I.B.: Problems of modeling natural and anthropogenic processes in the arctic zone of the Russian Federation. Math. Models Comput. Simul. **11**, 226–246 (2019)
3. Xunqiang, Y., Jianbo, L., Chenglin, W., Gao, L.: ANSYS implementation of damping solvent stepwise extraction method for nonlinear seismic analysis of large 3-D structures. Soil Dyn. Earthq. Eng. **44**, 139–152 (2013)
4. Nikolic, Z., Zivaljic, N., Smoljanovic, H., Balic, I.: Numerical modelling of reinforced concrete structures under seismic loading based on the finite element method with discrete inter element cracks. Earthq. Eng. Struct. Dyn. **46**(1), 159–178 (2017)
5. Moczo, P., Robertsson, J.O., Eisner, L.: The finite-difference time-domain method for modeling of seismic wave propagation. Adv. Geophys. **48**, 421–516 (2007)
6. Komatitsch, D., Tromp, J.: Introduction to the spectral element method for three-dimensional seismic wave propagation. Geophys. J. Int. **139**(3), 806–822 (1999)
7. Wilcox, L.C., Stadler, G., Burstedde, C., Ghattas, O.: A high-order discontinuous Galerkin method for wave propagation through coupled elastic-acoustic media. J. Comput. Phys. **229**(24), 9373–9396 (2010)
8. De Basabe, J., Mrinal, S., Wheeler, M.: The interior penalty discontinuous Galerkin method for elastic wave propagation: grid dispersion. Geophys. J. Int. **175**(1), 83–93 (2008)
9. Favorskaya, A.V., Zhdanov, M.S., Khokhlov, N.I., Petrov, I.B.: Modeling the wave phenomena in acoustic and elastic media with sharp variations of physical properties using the grid-characteristic method. Geophys. Prospect. **66**(8), 1485–1502 (2018)
10. Favorskaya, A.V., Breus, A.V., Galitskii, B.V.: Application of the grid-characteristic method to the seismic isolation model. In: Petrov I.B., Favorskaya A.V., Favorskaya M.N., Simakov S.S., Jain L.C. (eds.): Smart modeling for engineering systems. GCM50 2018. SIST, vol 133, pp. 167–181. Springer, Cham (2019)
11. Nikitin, I.S., Burago, N.G., Golubev, V.I., Nikitin, A.D.: Methods for calculating the dynamics of layered and block media with nonlinear contact conditions. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Theory and Practice of Computational Mechanics. SIST, vol. 173, pp. 171–183. Springer, Singapore (2020)
12. Golubev, V.I., Shevchenko, A.V., Petrov, I.B.: Taking into account fluid saturation of bottom sediments in marine seismic survey. Doklady Math. **100**(2), 488–490 (2019)
13. Muratov, M.V., Petrov, I.B.: Application of mathematical fracture models to simulation of exploration seismology problems by the grid-characteristic method. Comput. Res. Model. **11**(6), 1077–1082 (2019)
14. Petrov, I.B., Muratov, M.V.: Application of the grid-characteristic method to the solution of direct problems in the seismic exploration of fractured formations (review). Math. Models Comput. Simul. **11**, 924–939 (2019)

15. Grigorievih, D.P., Khokhlov, N.I., Petrov, I.B.: Calculation of dynamic destruction in deformable bodies. Matematicheskoe Modelirovanie **29**(4), 45–58 (2017)
16. Fedorenko, R.P.: A relaxation method for solving elliptic difference equations. USSR Comput. Math. Math. Phys. **1**(4), 1092–1096 (1962)

Chapter 14
Numerical Study on the Teeth Installation Parameters: Shift and Tilt Angle Effects

Sergey D. Arutyunov, Dmitry I. Grachev, Grigoriy G. Bagdasaryan, Ilia S. Nikitin, and Alexander D. Nikitin

Abstract The chapter is dedicated to the study on the teeth installation parameters on the stress state of the prosthesis under typical chewing loads. The two main parameters are investigated: the role of the dentition installation line and the role of tilt angle of teeth blocks. The simple 3D models were developed and used for these calculations. The physically based boundary conditions are introduced. The results of the calculation show a higher sensitivity of the lower prosthesis basis to vary the parameters compared to the upper prosthesis basis. It is shown that external shift of teeth installation line leads to higher stress intensity compared to oral one. The tilt angle effect results in slightly lower stress intensity compared to the shift effect. The oral tilt angle effect leads to higher stress intensity compared to the external tilt angles.

14.1 Introduction

In the recent years, the progress in computer-aided design and engineering leads to deep integration of mechanics and denture [1–5]. The problems of teeth reparation,

S. D. Arutyunov · D. I. Grachev · G. G. Bagdasaryan
A.I. Evdokimov Moscow State University of Medicine and Dentistry, 20/1, Delegatsksya ul., Moscow 127473, Russian Federation
e-mail: sd.arutyunov@mail.ru

D. I. Grachev
e-mail: dr.grachev@mail.ru

G. G. Bagdasaryan
e-mail: dr.bagdasaryan@mail.ru

S. D. Arutyunov · D. I. Grachev · G. G. Bagdasaryan · I. S. Nikitin · A. D. Nikitin (✉)
Institute for Computer Aided Design of the RAS, 19/18, Vtoraya Brestskaya Ul, Moscow 123056, Russian Federation
e-mail: nikitin_alex@bk.ru

I. S. Nikitin
e-mail: i_nikitin@list.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_14

implantation [6], shape, and position corrections [7] are needed for computer-aided simulations. Many high-level dental clinics are equipped by high performance system for measuring different parameters such as compressive force, occlusion, and so on. The recent progress in 3D techniques allows to reconstruct the geometry of dentation and profiles of the oral cavity by non-contact methods. Also, the progress in 3D techniques such as 3D scanning and 3D printing forced the dental specialists to develop the personalized solution in implantation, prosthetics, and teeth repairing. The personalized model has a quite complex shape close to the patient's native structure, but the result of a such calculation are strongly depended on chosen models of human soft tissues and corresponding boundary conditions. It is well known that physical and mechanical properties of the bones and tissues are quite different for different patients and cannot be correctly determined by non-destructive methods. Moreover, these properties can change in time, age, sex, and other individual parameters. Therefore, a general model of denture is requested for the determination of the main reaction of the prosthesis to the different geometry modifications. This model should have a simple shape and clear physically based boundary conditions. Such simple models are proposed for upper [8] and lower laminar prosthesis basis [9] with the physically based boundary conditions.

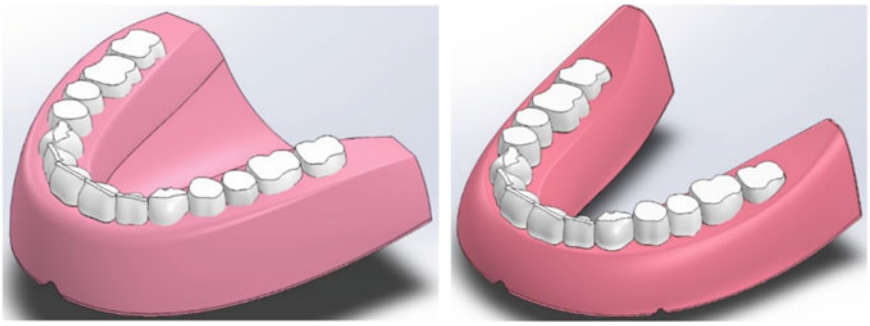
The chapter is organized as follows. Section 14.2 provides a mathematical modeling of laminar dentures of the upper and lower jaws. The boundary conditions and chewing loads are given in Sect. 14.3. Section 14.4 presents the results and discussions, while Sect. 14.5 concludes the chapter.

14.2 Mathematical Modeling of Laminar Dentures of the Upper and Lower Jaws

The simple models of upper and lower laminar prosthesis basis are presented in Fig. 14.1. The model consists of two parts: denture blade (in pink color) and dentition. The dentition is separated into four logical blocks: incisors, fang, premolars, and molars. The prosthesis blade is assumed to be made of homogeneous and isotropic material (Acryl plastic R) providing in Table 14.1. The thickness of the prosthesis base is contestant at all locations and equal to 1 mm. The teeth models have visible and hidden parts. The geometries for the dentition are taken from the work of Arutyunov [10]. The visible part is shown in Fig. 14.1 while the hidden part is integrated into the basis. The connection is realized by chemical bond that allows us to simulate this contact as a full grip. The material for the teeth model is also assumed homogeneous and isotropic with the following mechanical properties, Table 14.1.

The model of the upper prosthesis basis (Fig. 14.1a) has a shell-formed connection of parts covering the alveolar ridges. Both models have special technological notches that are used for cords pathing. The lower prosthesis basis has four such notches.

The poly methyl methacrylate acryl (PMMA) is the most common material used to fabricate the complete and partial dentures [11–13]. According to literature data



a b

Fig. 14.1 CAD models of lamellar bases in the case of complete absence of teeth: **a** upper jaw, **b** lower jaw

Table. 14.1 Mechanical properties of the acrylic plastics used for denture prosthesis

Material	Application	Mechanical properties	
Acrylic plastic *R*	Prosthesis basis	Young's modulus	1000 MPa
		Density	1000 kg/m^3
		Poisson ratio	0.3
Acrylic plastic *W*	Dents	Young's modulus	2000 MPa
		Density	1000 kg/m^3
		Poisson ratio	0.3

[14], the ultimate tensile strength (UTS) for PMMA materials varies from 47 to 79 MPa. The average value of UTS in the present study is taken equal to 60 MPa.

14.3 Boundary Conditions and Chewing Loads

The boundary conditions are developed based on the typical morphology of the oral cavity. The structure of bones and soft tissues are different for the upper and lower jaws. The outstanding feature of the upper jaw is the area of the palatine bones connection or "torus," Fig. 14.2b. The area of the torus is characterized by a lower compliance compared to neighboring regions [15]. The area of the torus is located on the prosthesis symmetry and has an oval shape, Fig. 14.2a. The reaction of tissue is simulated as distributed pressure with the minimum compliance at the geometrical center of the ellipse with following increasing to the edge. At the ellipse edge, the compliance of tissue is taken equal to the parameters of the rest surface. The value of compliance is taken from the experimental data presented by Kulagenko [15].

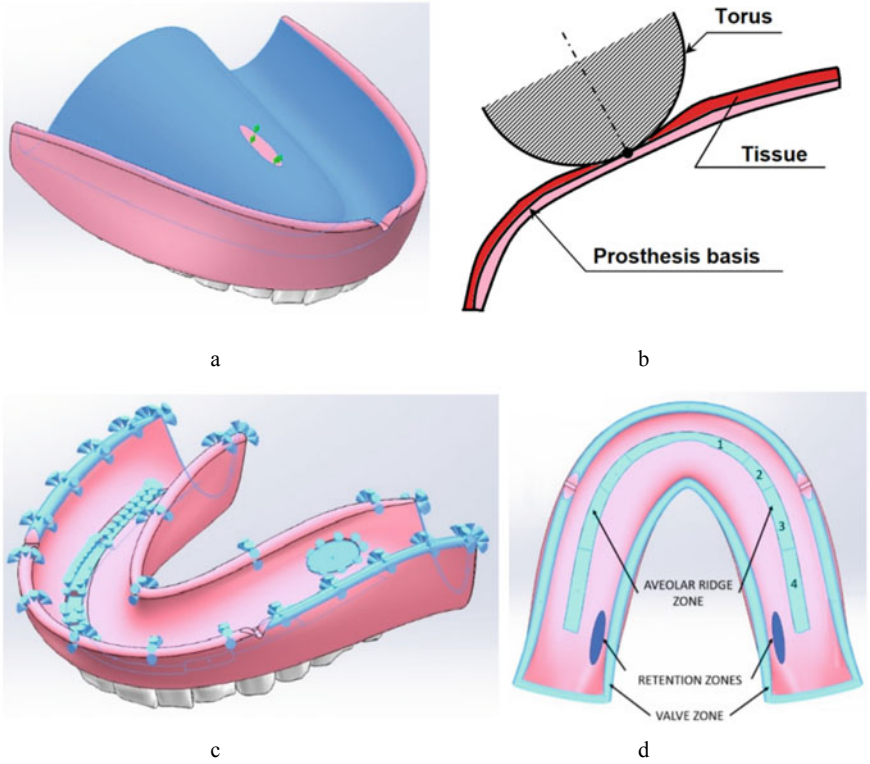


Fig. 14.2 3D models of: **a** upper prosthesis basis, **b** lower prosthesis basis, **c** with boundary conditions of upper prosthesis basis, **d** with boundary conditions of lower prosthesis basis (due to contact with soft tissues of oral cavity)

In the case of the lower part of the oral cavity, the structure of soft tissue is different. The main part of the load is carried by the alveolar ridges. The compliance of these ridges is lower compared to the rest surface, Fig. 14.2d. However, the compliance of these areas is about 20% higher compared to torus area. The compliance distribution is not homogeneous along the ridge line. In order to simulate that, the whole area of alveolar ridge was separated into four zones. These zones are assumed having the same properties for the left and right branches of the ridge. The compliance distribution was estimated based on the hypothesis about the horizontal position of the prosthesis under uniformly distributed normal load. At the first stage of the calculations, all the teeth were loaded by the normal pressure of a given value. The compliance distribution was assumed also homogeneous. The result of calculation has shown different vertical displacement of the tail and face parts of the prosthesis. Further, the compliances within the four sections were modified in order to rich the horizontal position for the basis under uniformly distributed loads.

The second feature of the lower prosthesis basis is significant role of valve zone, Fig. 14.2d. The physics of this valve zone appearance is related to adhesive forces acting between tissue and prosthesis basis in the presence of viscous fluid (saliva). The normal adhesive forces are assumed to be homogeneously distributed along the prosthesis perimeter. These forces should be taken into an account during the vertical displacement calculations.

The last outlining zone is retention zones that are located at the left and right branches of the alveolar ridge. These zones are related to typical geometry of lower jaw bone. The nature of these forces is similar to friction forces. The force is acting in opposite direction of the local displacements. These boundary conditions are corresponding to the main physical aspects of lower prosthesis behavior and used for the present calculations.

When simulating a chewing load, a complete cycle of biting and chewing food is reproduced. For this purpose, four separate tooth blocks were identified as block I (incisors), block II (canine), block III (premolars), and block IV (molars) (see Fig. 14.3).

It is assumed that the maximum load is determined by the amount of muscle effort, taken equal to 100 N. Further, the magnitude of muscle effort is converted into the value of pressure acting on the corresponding tooth blocks. The load can be as symmetrical, Fig. 14.3a, as well asymmetric Fig. 14.3b. In this case, the pressure value, as before, is calculated keeping constant the muscle effort. The block separation was the same for the upper and lower jaw. The corresponding pressure values are given in Table 14.2.

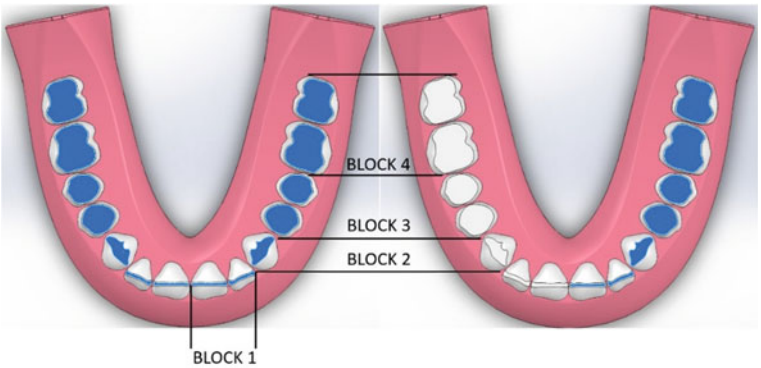


Fig. 14.3 Blocks of teeth: block 1 (incisors), block 2 (canine), block 3 (premolars), and block 4 (molars) to which the load is applied

Table. 14.2 Loading parameters for symmetrical and non-symmetrical loadings

Loading type/location	Block I	Block II	Block III	Block IV
Symmetrical loading, pressure, MPa	5.2	4.2	0.9	0.5
Non-symmetrical loading, pressure, MPa	10.4	8.4	1.8	1.0

14.4 Results and Discussions

The aim of the research is to study an influence of teeth role parameters (installation line and tilt angle) on the stress distribution and structural integrity of prosthesis basis under typical chewing loads. According to the medicine practice, the teeth installation line can be shifted to the oral or external sides, see Fig. 14.4a, b. The common practice is to adjust the position of individual teeth blocks to the individual

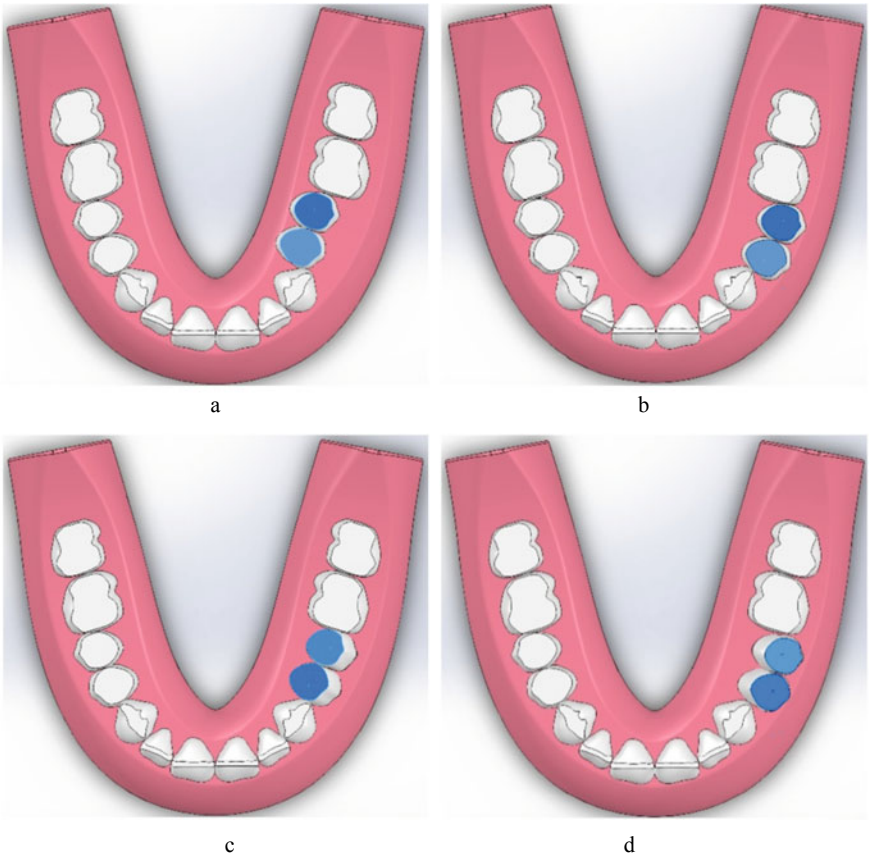


Fig. 14.4 3D model with: **a** oral shift, **b** external shift of teeth installation line, **c** oral tilt angle, **d** external tilt angle of the teeth block for the lower prosthesis

features of oral cavity. However, there are quite limited number of studies on the permissible range for the teeth block shift and tilt angles. According to the basics of mechanics, the most critical case for the structural integrity of the prosthesis basis is the loads applied to the shift teeth blocks or on the block with a tilt angle. In the framework of this study, all the configurations are studied: oral and external teeth block shift (from block 1 to block 4) and oral and external tilt angles for these four blocks. The maximum teeth shift is ±1 mm from the normal installation line. The tilt angle is varying from –20° to 20°. The results for extreme values are given.

Hereinafter, the studies on the shift effect and tilt angle effect are presented in Sects. 14.4.1–14.4.2, respectively.

14.4.1 Study on the Shift Effect

The results of the external shift effect are presented in Fig. 14.5. Due to small contact area of the block 1 and block 2, the critical cases are corresponding to the shifts of these blocks. Figure 14.5 contains the results of calculation for the block 1 and block 2. The absolute value of applied pressure at a given block is given in Table 14.1.

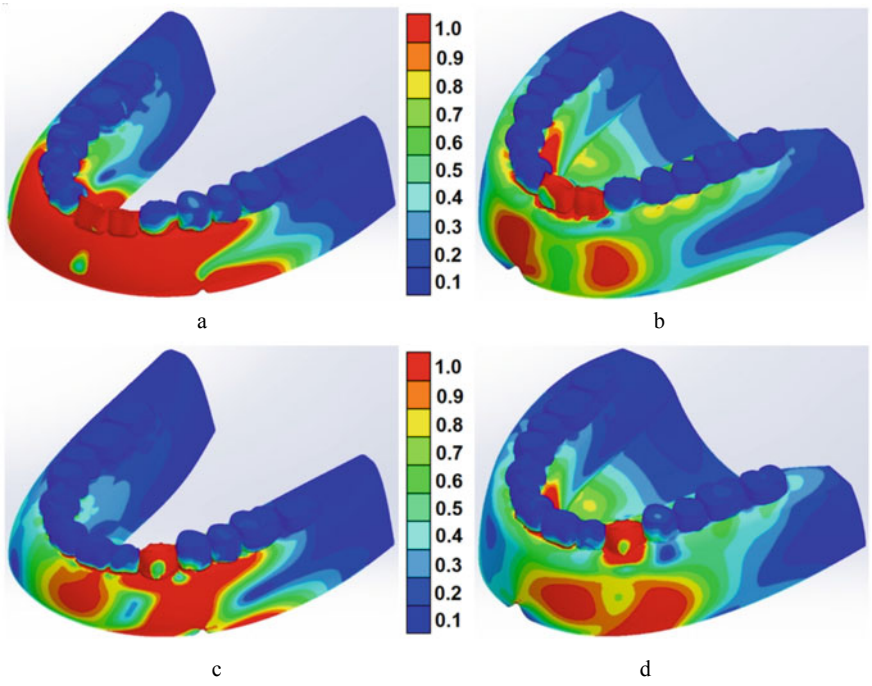


Fig. 14.5 3D model with external shift of teeth installation line for: **a** block 1 at lower prosthesis, **b** block 1 at upper prosthesis, **c** block 2 at lower prosthesis, **d** block 2 at upper prosthesis

The results of the calculations are presented for only the critical cases. The stress field is normalized by the maximum stress level found for a given configuration. Based on the obtained results, the external shift of teeth installation line is more critical for the case of lower prosthesis basis. The area of elevated stress is localized at the face part of the prosthesis. The area of high stress is vast, and the significant stress level can be found up to molars. In the case of upper prosthesis, the areas of elevated stress are fragmental. The local area can be found at the tip of the artificial notch and at the location of first block. A general stress distribution is less intense at the face part of the prosthesis.

In the case of loading at the block 2, the stress state is comparable for lower and upper prosthesis basis. The stress distribution has a fragmentary shape with local maximums. For the both cases, the high stress level is found at the teeth of block 2 and its vicinity. In the case of loads at block 2, the stress state for the lower prosthesis basis is still higher.

The similar results were obtained for the case of oral shift of the teeth installation line, Fig. 14.6. In the case of oral shift of the teeth installation line, the stress intensity is lower compared to external case. The locations of maximum stress remain similar.

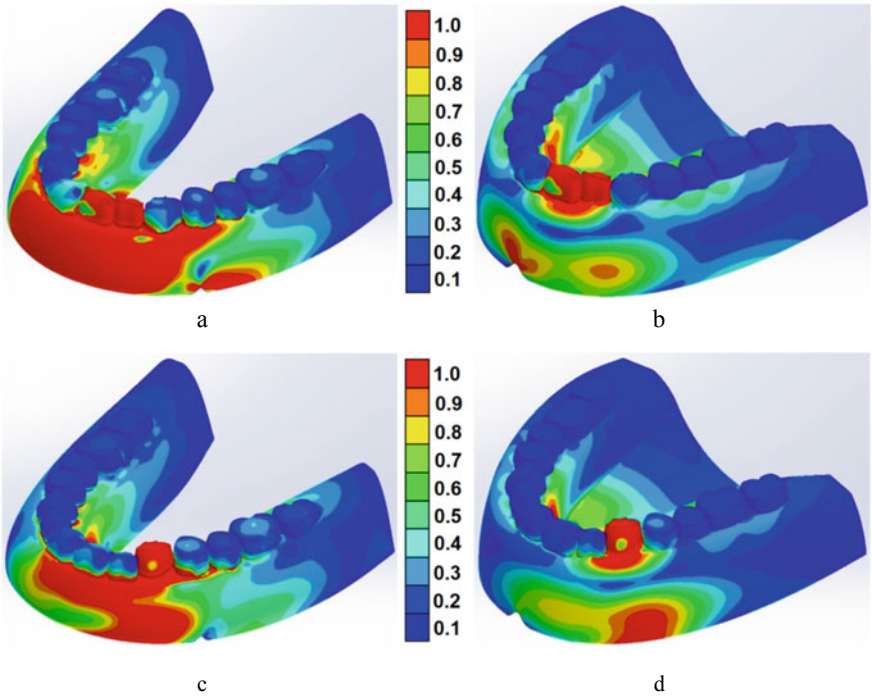


Fig. 14.6 3D model with oral shift of teeth installation line for: **a** block 1 at lower prosthesis, **b** block 1 at upper prosthesis, **c** block 2 at lower prosthesis, **d** block 2 at upper prosthesis

The results of shift effect show the critical role of external teeth displacement. The found feature is more pronounceable for the case of lower prosthesis basis. In the case of upper prosthesis basis, the stress distribution is fragmentary with local maximums at the teeth vicinity or artificial notches.

14.4.2 Study on the Tilt Angle Effect

The same calculations were performed for the case of teeth blocks that are located at the normal installation line but have a tilt angle. There are two configurations: oral and external tilt angles as shown in Fig. 14.4c, d. The applied pressures are listed in Table 14.1 for the corresponding teeth block. The results of calculations are presented in Figs. 14.7 and 14.8.

The results of calculation for the configurations with the oral tilt angle are presented in Fig. 14.7. The stress level bars are normalized by the maximum stress value calculated for a given configuration. The stress field is more intense in the case of lower prosthesis basis. The stress distribution for the upper prosthesis basis is more discreet with local maximum at the loaded teeth locations.

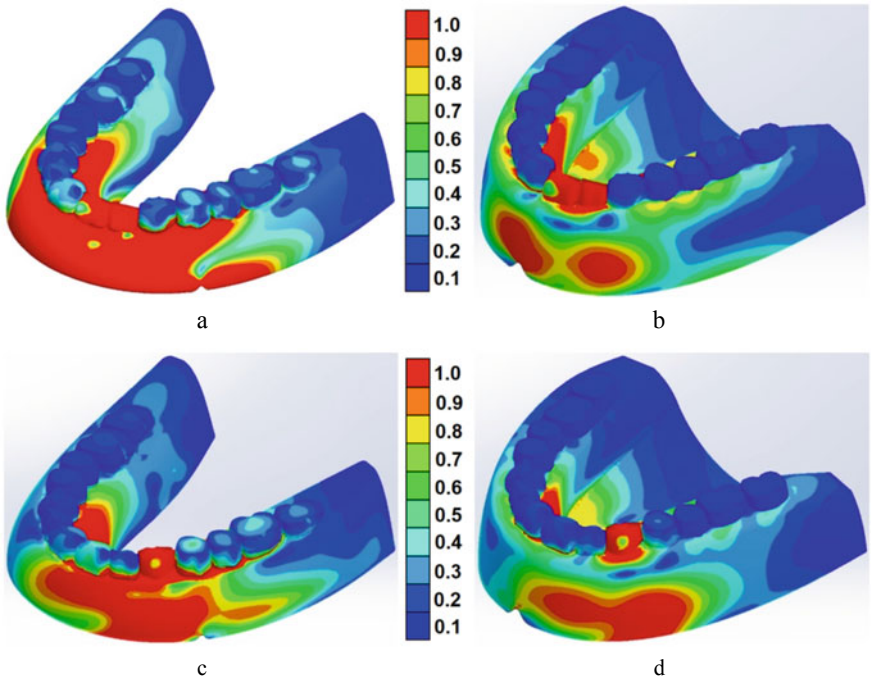


Fig. 14.7 3D model with oral tilt angle of teeth installation line for: **a** block 1 at lower prosthesis, **b** block 1 at upper prosthesis, **c** block 2 at lower prosthesis, **d** block 2 at upper prosthesis

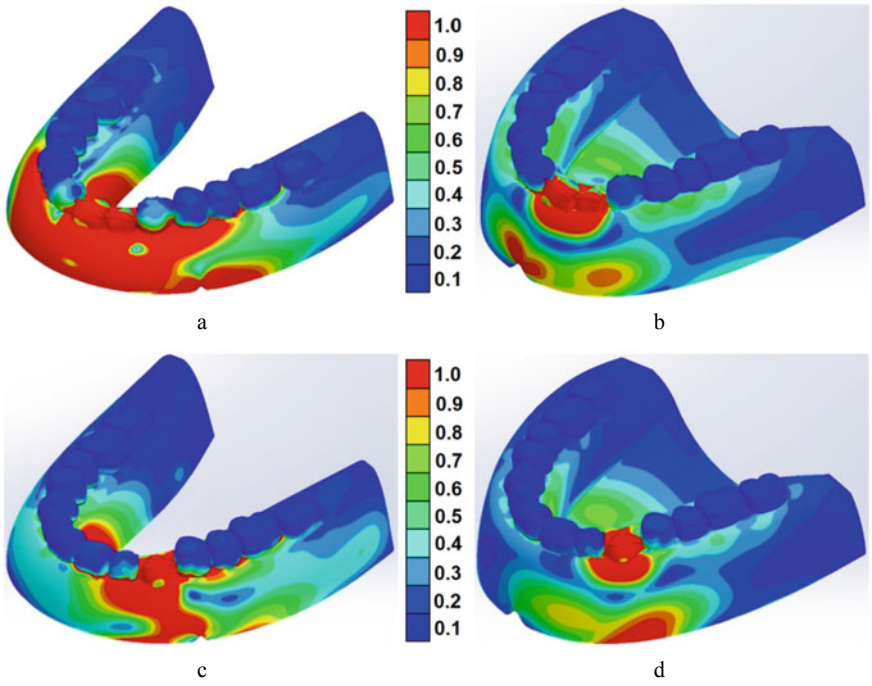


Fig. 14.8 3D model with external tilt angle of teeth installation line for: **a** block 1 at lower prosthesis, **b** block 1 at upper prosthesis, **c** block 2 at lower prosthesis, **d** block 2 at upper prosthesis

The stress intensity is lower for the case of oral tilt angle compared to oral shift of teeth installation line. In the case of loading at block 2, the stress intensity is lower compared to the block 1 configuration. The stress distribution is discreet with local maximum at the loaded teeth locations. The lower prosthesis basis has higher stress intensity.

The results for the external tilt angle are presented in Fig. 14.8. The found tendency for the oral tilt angle configurations keeps the same for the external case. However, the stress intensity is lower, compared to the previous problem. Thus, the oral tilt angle configuration is more critical for structural integrity of the prosthesis basis compared to the external one. However, the role of title angles is lower compared to the shift of teeth installation line.

14.5 Conclusions

The simple models of upper and lower laminar prosthesis basis are presented for the applied problems. The physically based boundary conditions are introduced. The role of teeth installation line shift and tilt angle is studied. It is shown that the loads at the

block 1 and block 2 locations can be critical for the structural integrity of prosthesis basis. The study on the shift effect and tilt angle show the higher sensitivity of the lower prosthesis basis to these parameters. The external shift of teeth installation line is more critical compared to oral shift. In the case of tilt angle investigation, the same tendency was found, i.e., the lower prosthesis basis has the higher sensitivity to the parameters. However, the oral tilt angle leads to higher stress intensively. These results can be used in medical practice during the individual strategy developing for the teeth installation. The presented models can be used for the approbation of a general hypothesis and ideas in dentistry.

Acknowledgements The work is realized in the framework of the state contract of the Institute for Computer-Aided Design of the RAS. The authors are grateful to A. Zhuravlev for the significant contribution to the present work.

References

1. Weisgeim, L.D., Scherbakov, L.N., Goncharov, A.A.: Influence of separate clinical aspects on the tensely deformed condition of the biomechanical system «combined prosthesis—supporting tissues». Stomatologia **4**(16), 18–20 (2009)
2. Iskanderov, R.M., Gvetadze, R.S., Butova, V.G., Andreeva, S.N., Timofeev, D.E.: The overall development strategy of dental laboratories equipped with CAD/CAM-systems. Stomotologia **98**(2), 8–12 (2019)
3. Chumachenko, E.N., Arutyunov, S.D., Lebedenko, I.Y., Ilinich, A.N.: Analysis of load distribution and the likelihood of irreversible changes in the jaw bone tissue during orthopedic treatment using dental intraosseous implants. Clin. Dent. **2**, 1–44 (2002)
4. Al-Ali, M.A., Al-Ali, M.A., Takezawa, A., Kitamura, M.: Topology optimization and fatigue analysis of temporomandibular joint prosthesis. World J. Mech. **7**(12), 323–339 (2017)
5. Dubova, L.V., Tsarev, V.N., Zolkina, Y.S., Malik, M.V., Nikitin, I.S., Chuev, V.P.: Comparative assessment of milled materials for temporary unremovable dentures supported by the isoelastic implants according to the experimental study of their stress-strain states and microbial adhesion. Clin. Dent. **3**(87), 74–78 (2019)
6. Perelmuter, M.N.: Analysis of stress-strain state of dental implants by boundary integral equations method. PNRPU Mech. Bull. **2**, 83–95 (2018)
7. Shanidze, Z.L., Muslov, S.A., Arutyunov, A.S., Astashina, N.B., Arutyunov, S.D.: Biomechanical approach to dental orthopedic treatment of patients with postoperative maxillary defect. Russian J. Biomech. **24**(1), 28–38 (2020)
8. Aruyunov, S.D., Grachev, D.I., Nikitin, A.D.: Mathematical modelling on the fracture of a laminar prosthesis basis under natural chewing loads. In: IOP Conference Series Materials Science and Engineering, vol. 747, pp. 012065.1–012065.6 (2020)
9. Arutyunov, S.D., Grachev, D.I., Bagdasaryan, G.G., Nikitin, A.D.: Critical stress analysis for the basis of a denture prosthesis. In: IOP Conference on Series Materials Science and Engineering (2020). (In print)
10. Arutyunov, S.D., Chumachenko, E.N., Lebedenko, I.Y., Arutyunov, A.S.: Comparative analysis of the mathematical modeling results on the stress-strain state of various designs for pin dentures. Dentistry **2**, 1–41 (2001)
11. Abdulrazzaq, N.S., Jafarzadeh, K.T., Behroozibakhsh, M., Hajizamani, H., Hajizamani, S.: Recent advances and future perspectives for reinforcement of poly(methyl mathacrylate) denture base materials: a literature review. J. Denture Biomaterials **51**, 490–502 (2018)

12. Oguzhan Y., Melik, S., Kemal, G.U.: Biocompability of dental polymers. In: Méndez-Vilas, A., Solano, A., (eds) Polymer Science: Research Advances, Practical Applications and Educational Aspects, pp. 89–98. Formatex Research Center S.L., Spain (2016)
13. Gautam, R., Singh, R.D., Sharma, V.P., Siddhartha, R., Chand, P., Kumar, R.: Biocompatibility of polymethylmethacrylate resins used in dentistry. J. Biomed. Mater. Res. B Appl. Biomater. **100**(5), 1444–50 (2012)
14. Oleiwi, J.K., Hamad, Q.A.: Studying the mechanical properties of denture base materials fabricated from polymer composite materials. Al-Khwarizmi Eng. J. **14**(3), 100–111 (2018)
15. Kulazenko, V.I., Berezovskiy, S.S.: Clasp prosthetics. K.: Zdorovie, **103**, (1975)

Part IV
Numerical Study of Dynamic Systems

Chapter 15
Astronomical and Geophysical Factors of the Perturbed Chandler Wobble of the Earth Pole

Sergej S. Krylov, Vadim V. Perepelkin, and Alexandra S. Filippova

Abstract In the framework of the restricted three-body problem, a celestial–mechanical model of the steady-state Chandler wobble of the Earth pole is proposed. The contribution of the astronomical and geophysical disturbances to the observed Earth pole oscillations is discussed based on the processing of IERS observations of the Earth pole motion, NCEP/NCAR geophysical data of the atmospheric circulation, and NASA/JPL angular momentum of the ocean. The directions of the axes x', y' corresponding to 50° of west longitude and 40° of east longitude, respectively, are found in the projection, onto which its coordinates have the maximum and minimum intensities of perturbed oscillations. The Earth pole oscillatory process that is in-phase with the lunar orbit precessional motion is studied, and the contribution of moving media to this process is discussed.

15.1 Introduction

The study of the fundamental astrometric problem of predicting the Earth pole motion [1] is of significant theoretical interest and fundamental for satellite navigation [2, 3]. One of its important problems is a high-precision forecast of the spacecraft orbits [4, 5]. In order to solve this problem, it is necessary to take into account various perturbing factors in the equations of motion [3, 6]. Accuracy improvement of the coordinate-time and navigation support of the satellite systems is closely related to the prediction of the Earth pole oscillations because, for example, the Earth orientation

S. S. Krylov · V. V. Perepelkin (✉) · A. S. Filippova
Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, Moscow 125993, Russian Federation
e-mail: vadimkin1@yandex.ru

S. S. Krylov
e-mail: krylov@mai.ru

A. S. Filippova
e-mail: filippova.alex@gmail.com

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_15

parameters are included in the transformation matrix from the geocentric equatorial coordinate system to the Earth geographic coordinate system. One of the main problems in predicting the Earth pole motion is to take into account the parameters variability in the main components of the Earth pole oscillation (Chandler wobble and annual oscillatory process) [7].

Usually, the Chandler wobble is understood as the Earth pole oscillation with the frequency of free nutation of the Earth's rotation axis (with the Chandler frequency) in the Earth-bound coordinate system [8]. It can also be considered in a narrower sense, as a steady-state oscillation mode at the Chandler frequency and, in a broader sense, as a multi-frequency oscillatory process with a main frequency being close to Chandler's one. However, it is very difficult to give an unambiguous definition that would fully correspond to the physical process under consideration. Uncertainty in the interpretation of the Chandler component is due to the lack of a comprehensive explanation of the excitation mechanism. In some cases, it is convenient to use the terminology of the perturbation theory. If the steady-state regime of the Chandler wobble (with a constant frequency and average amplitude) is formally taken as an "unperturbed" motion, then the perturbations that lead to variations in the Chandler wobble parameters, which may be considered as perturbations, although it should be noted that the steady state of the Chandler wobble is also a perturbed motion.

Explaining the excitation mechanism of the Chandler wobble is one of the fundamental problems when studying the Earth pole motion. At least, a part of the perturbations leading to variations in the Chandler wobble parameters are caused by this mechanism. Therefore, the study of the variability of the main components parameters of the Earth pole oscillation (generally speaking, not only Chandler wobble, but also annual oscillations) is of considerable interest both for the task of predicting the Earth pole motion and the study of the excitation mechanism of the Chandler wobble. First of all, the problem is to identify the celestial–mechanical and geophysical reasons for such behavior of the Chandler component of the Earth pole oscillations.

Factors affecting the Earth motion relative to the center of mass can be divided into astronomical and geophysical. The Earth motion in space, as well as, the motion of the Earth's moving media occurs under the influence of the bodies in the solar system, and primarily under the Sun and the Moon. Therefore, when studying the Earth motion it is natural and necessary to take into account the astronomical and geophysical factors together. Lunar–solar gravitational perturbing forces lead to the precession and nutation of the Earth, the refined theory of which taking into account the internal structure of the Earth is in good agreement with observational data. In contrast to precession and nutation, the Earth deformability and the mobility of its various media are determining factors for the motion of the instantaneous axis of rotation in the Earth's body. And in this case, it is important to take into account not only the mobility of the media, but also the astronomical factors that influence them, since during the evolution of the solar system many processes must be considered as synchronized processes.

The purpose of this chapter is to study the oscillatory processes of the Earth pole under the perturbing astronomical and geophysical factors, as well as, the aspects of their synchronization. In Sect. 15.2, the definition of the unperturbed Earth pole

motion is introduced based on the celestial–mechanical model of the deformable Earth rotation. Tidal oscillations in the inertia tensor of a deformable Earth, which are taken into account in the framework of a simple celestial–mechanical model of its motion, are considered in Sect. 15.3. In Sect. 15.4, a correspondence between the intensity of perturbed oscillations in the Earth pole coordinates, the direction of the coordinate axes and the longitude distribution of the ocean surface is established bassed on the processing of astrometric and geophysical observation data. Section 15.5 is devoted to the study of the geophysical disturbances contribution to the synchronization between the Earth pole motion and precession of the lunar orbit. In Sect. 15.6, the main conclusions of the work are given.

15.2 Studying the Earth Rotation Within the Restricted Three-Body Problem

The study of the Earth motion relative to its center of mass under the lunar–solar gravitational–tidal and geophysical disturbances is based on the problem of a system consisting of a deformable planet (the Earth) and a point satellite (the Moon) moving around an attracting center (the Sun) [9–12]. The Earth and the Moon perform translational–rotational motion around the barycenter, which moves in orbit around the Sun (Fig. 15.1).

We introduce the inertial coordinate system $O\xi_1'\xi_2'\xi_3'$ with the origin in the attracting center O, where the axis $O\xi_3'$ is orthogonal to the orbital plane of the

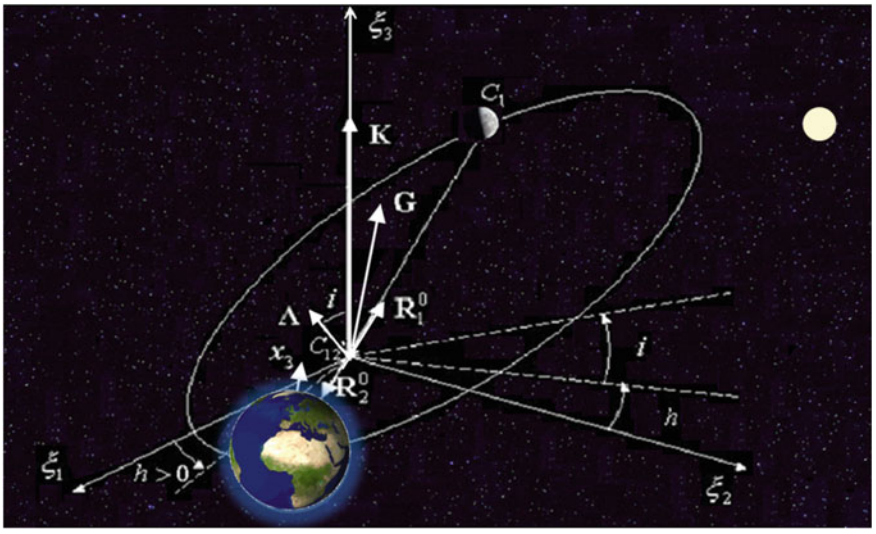


Fig. 15.1 Coordinate system for the two-body problem and orientation of the vectors

barycenter C_{12}, and the Koenig coordinate system $C_{12}\xi_1\xi_2\xi_3$. In an undeformed state, the Earth is dynamically compressed, i.e., $C > A$, where C and A are the axial and average equatorial moments of inertia, respectively. Let us bind the coordinate system $C_2'x_1x_2x_3$ with the deformable Earth, in a way that the axes are directed along the main central axes of the undeformed planet and the point C_2' is the center of mass of the planet in the absence of deformations.

Let $\mathbf{G}$ be the spin of the planet, $\mathbf{\Lambda}$ be the orbital angular momentum of the satellite's centers of mass C_1 and the planet's C_2. In the absence of disturbances, the angular momentum of the system $\mathbf{K} = \mathbf{G} + \mathbf{\Lambda}$ is stationary in inertial space and coincides with the $C_{12}\xi_3$ axis (Fig. 15.1).

The deformable Earth motion relative to the center of mass can be described in the canonical variables of Andoyer (Fig. 15.2) $L,\ G,\ G_{\xi_3},\ \varphi_1,\ \varphi_2,\ \varphi_3$, ($G = |\mathbf{G}|$, L is a projection of the vector $\mathbf{G}$ on the axis $C_2'x_3$, and G_{ξ_3} is a projection of the vector $\mathbf{G}$ on the $C_{12}\xi_3$). We describe the mutual orbital motion of the centers of masses C_1 and C_2 in the Delaunay canonical variables $\Lambda,\ H,\ \vartheta,\ h$ ($\Lambda = |\mathbf{\Lambda}|$, H is the projection of the vector $\mathbf{\Lambda}$ on the $C_{12}\xi_3'$ axis, ϑ is the mean anomaly, and h is the longitude of the ascending node of the orbit on the $C_{12}\xi_1\xi_2$ plane).

In the bounded coordinate system, the unit vectors $\mathbf{R}_{21}^0$ and $\mathbf{R}^0$, which specify the directions from the Earth to the Moon and from the Sun to the barycenter, respectively, are defined as follows equations:

$$O^{-1}(t)\mathbf{R}_{21}^0 = (\gamma_1, \gamma_2, \gamma_3)^T,$$
$$O^{-1}\mathbf{R}^0 = (\kappa_1, \kappa_2, \kappa_3)^T.$$

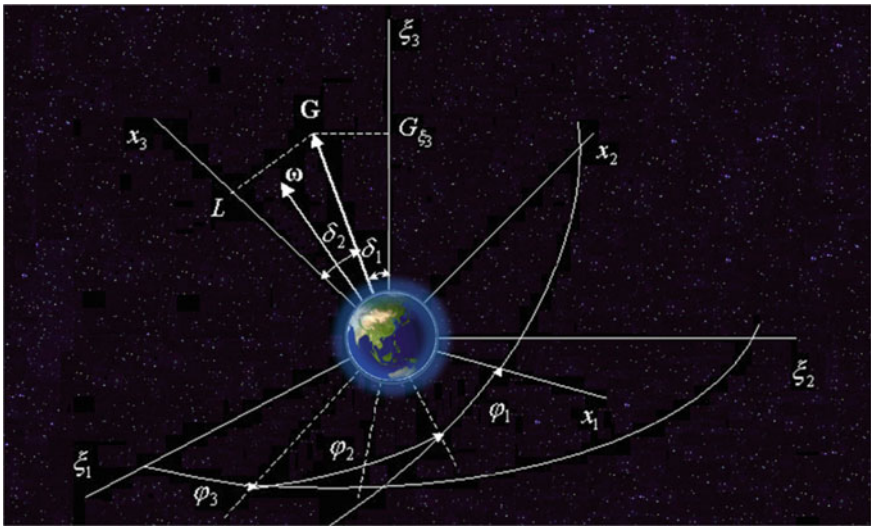


Fig. 15.2 Mutual orientation associated with the deformable Earth and reference coordinate systems in Andoyer variables

For practical applications, the transformation between two geocentric coordinate systems is important—the Konig one $C_2\xi_1\xi_2\xi_3$ and the Earth-bound one $C_2x_1x_2x_3$. This conversion is carried out by five consecutive rotations at the angles $\varphi_1,\ \delta_2,\ \varphi_2,\ \delta_1,\ \varphi_3$ according to Eq. 15.1.

$$O^{-1}(t) = \Gamma_3^{-1}(\varphi_1)\Gamma_1^{-1}(\delta_2)\Gamma_3^{-1}(\varphi_2)\Gamma_1^{-1}(\delta_1)\Gamma_3^{-1}(\varphi_3) \tag{15.1}$$

The matrix $O(t)$, which defines the transition from the Earth-bound to inertial axes, is expressed in canonical Andoyer variables, and $\cos\delta_1 = G_{\xi_3}/G$, $\cos\delta_2 = L/G$, (Fig. 15.2). The last two angles δ_1 and φ_3 in transformation (Eq. 15.1) are determined by the precession and nutation of the Earth and for this study are considered known and given. The angles φ_1 and δ_2 are the polar coordinates of the Earth pole and the variations of the angle φ_2, which are associated with the irregularities of the Earth rotation, lead to variations of Universal Time UT1 [1].

The values of the pole shift and variations of Universal Time are very small: they do not exceed $0.5''$ for the annual Earth pole motion and 0.03 s for the annual amplitude of Universal Time variations. Changes in the angles of $\varphi_1,\ \delta_2,\ \varphi_2$ are significantly affected by the Earth deformations. The determination of variations in the inertia tensor of the deformable Earth is necessary to calculate the vector of the angular momentum, as well as, its total derivative by time, which is used to study both the perturbed and unperturbed Earth motion relative to its center of mass.

The most convenient generalized coordinates to qualitative describe the Earth's rotation around its center of mass are the canonical action-angle variables. The variables $I_1 = L,\ I_2 = G,\ I_3 = G_{\xi_3}, \varphi_1, \varphi_2, \varphi_3$ are the action-angle variables in the dynamically symmetrical Earth case.

For a qualitative description of the Earth motion relative to its center of mass, when taking into account the impact of disturbances from the Moon and the Sun, the linear theory of viscoelasticity of small deformations is used. The perturbed Routh functional of the problem under consideration can be represented in the form of Eq. 15.2 [9].

$$R = R_0 + \varepsilon R_1(\{I\}, \{\varphi\}, [\mathbf{u}], [\dot{\mathbf{u}}]) + \varepsilon^2 \ldots \tag{15.2}$$

Here, R_0 is the Routh functional in the absence of deformations including the functionals of the system's kinetic energy and potential energy of gravitational forces from the Moon and the Sun, εR_1 is the perturbation functional due to gravitational tides that includes the kinetic energy of the relative motions of the elastic body particles and potential energy of elastic deformations, $\mathbf{u}$, $\dot{\mathbf{u}}$ are vectors of displacement and velocity of the moving medium particles; $\varepsilon > 0$ is a small dimensionless parameter characterizing the relative magnitude of the perturbing factors in Eq. 15.2, which is introduced for convenience.

The dynamics of the perturbed Chandler motion of the instantaneous axis is related, in particular, with a change in the angle δ_2, which determines the change in the amplitude of the Chandler wobble. The angular variable δ_2 is the angle between the axis of the figure of the Earth and the vector of the Earth's spin.

In the absence of dynamic symmetry ($A \neq B$), the action-angle variables will differ from the Andoyer variables by small changing values, and the desired equations taking into account the perturbed functional εR_1 will take the form of Eqs. 15.3, where h_1 is the integral of the kinetic energy for the unperturbed problem.

$$\begin{gathered}
\dot{I}_1 = -\varepsilon \frac{\partial R_1}{\partial w_1} \qquad \dot{I}_2 = 0 \qquad \dot{I}_3 = -\varepsilon \frac{\partial R_1}{\partial w_3} \\
\dot{w}_1 = n_1 + \varepsilon \frac{\partial R_1}{\partial I_1} \qquad \dot{w}_2 = n_2 + \varepsilon \frac{\partial R_1}{\partial I_2} \\
\dot{\delta}_2 = -\varepsilon (I_2 \kappa_* \sin \delta_2)^{-1} \frac{1+\kappa^2 \mathrm{sn}^2(\eta,\lambda)}{\mathrm{dn}(\eta,\lambda)} \frac{\partial R_1}{\partial w_1} \\
\eta = \frac{2}{\pi} \mathbf{K}(\lambda) w_1 \qquad \kappa^2 = \frac{C(A-B)}{A(B-C)} \qquad \lambda^2 = \kappa^2 \frac{2Ch_1 - I_2^2}{(I_2^2 - 2h_1 A)}
\end{gathered} \tag{15.3}$$

The model of the Chandler pole motion with the frequency $\dot{w}_1 = n_1$ and identification of its parameters are based on the qualitative theory of dissipative systems. To determine the steady pole motion as an unperturbed motion, the dissipative terms of the pole tide are taken into account in the variations of the centrifugal moments of inertia δJ_{pr}, δJ_{qr}. To do this, the Routh functional $\mathcal{R}_{01}$ of the perturbed problem is introduced as Eq. 15.4.

$$\mathcal{R}_{01} = -L\sqrt{G^2 - L^2}\left(\frac{\delta J_{\mathrm{pr}} \sin l}{A\,C} + \frac{\delta J_{\mathrm{qr}} \cos l}{B\,C}\right) \tag{15.4}$$

Variations in the centrifugal moments of inertia due to variable rotational deformation are associated with variations in the tesseral harmonics of the geopotential with simple relations [10]. The amplitudes of the variable normalized tesseral harmonic coefficients $\delta\overline{c}_{21}, \delta\overline{s}_{21}$ are determined from geophysical measurements and, according to [1], are related to the coordinates of the Earth pole by the relations:

$$\begin{bmatrix} \delta\overline{c}_{21} \\ \delta\overline{s}_{21} \end{bmatrix} = -1.33 \cdot 10^{-9}\left(\begin{bmatrix} x_p \\ y_p \end{bmatrix} + 0.0115 \begin{bmatrix} y_p \\ -x_p \end{bmatrix}\right).$$

Taking these terms into account, Eq. 15.4 leads to the damping of the pole motion at a frequency of n_1.

The perturbed motion taking into account the dissipative properties of the Earth viscoelastic mantle leads to regular precession with slowly changing parameters, which can be studied on the basis of asymptotic methods of nonlinear mechanics [11, 12]. And, in particular, the steady state of the Chandler wobble is determined.

When considering the perturbed case for $\mathcal{R}_{01}$ taking into account the dissipative terms of the pole tide in the variations of the centrifugal moments of inertia, as well as, the small perturbation at the Chandler frequency n_1 in a form of

$$\begin{aligned}
\frac{\delta J_{\mathrm{pr}}}{A^*} &= -\sigma \delta_2 \sin w_1 + \mu_p \cos(n_1 t + \beta_p), \\
\frac{\delta J_{\mathrm{qr}}}{B^*} &= -\sigma \delta_2 \cos w_1 + \mu_q \cos(n_1 t + \beta_q),
\end{aligned} \tag{15.5}$$

we obtain Eq. 15.6 for coefficient δ_2.

$$\dot{\delta}_2 \approx -\frac{2r_0 K(\lambda)\kappa}{\pi\chi}\sigma\delta_2 + f_p\sqrt{1+\kappa^2}\sin w_1 \cos(Nt+\beta_p) + f_q \cos w_1 \sin(Nt+\beta_q) \tag{15.6}$$

Here, σ is the dissipation coefficient, $f_{p,q}$ and $\beta_{p,q}$ are the amplitudes and phases of the perturbation, respectively.

In stationary steady state, we will have Eq. 15.7.

$$\begin{gathered}\delta_2 \approx \frac{f_q \sin(\beta_q+\Delta\psi)+f_p\sqrt{1+\kappa^2}\cos(\beta_p+\Delta\psi)}{4r_0 K(\lambda)\kappa\sigma(\pi\chi)^{-1}} \\ f_q\cos(\beta_q+\Delta\psi) - f_p\sin(\beta_p+\Delta\psi) = 0\end{gathered} \tag{15.7}$$

To study the dynamics of the Earth pole motion, the steady-state mode of its oscillations can be taken as unperturbed. Factors that perturb the steady motion of the Earth pole are astronomical (lunar–solar disturbances) and geophysical ones. The obtained model of the Earth pole unperturbed oscillatory process is also convenient for constructing a numerical–analytical model for predicting its motion [13].

15.3 Tidal Oscillations of the Deformable Earth Inertia Tensor

Modern methods of gravimetry, geophysics, and space geodesy make it possible to measure with high accuracy the temporal variations of the geopotential expansion coefficients and the corresponding small radial vibrations of the Earth's surface. These fluctuations occur mainly due to the lunar–solar tidal disturbances and geophysical phenomena. For example, the amplitudes of solid-state tides from the Moon and the Sun measured on the Earth's surface reach 34 and 16 cm, respectively. The magnitudes of these amplitudes are in accordance with the magnitudes of the equipotential surface oscillation amplitudes of the tidal potential and, to a first approximation, are connected by a linear dependence. The proportionality coefficient between the surface level heights of the tidal potential and the Earth's surface is determined from observations. It is associated with many physical and mechanical characteristics of the deformed Earth. The lunar–solar tidal potential leading to terrestrial tides—solid, oceanic, and atmospheric—also turns out to be proportional to the corresponding changes in geopotential. The estimate value of these proportionality coefficients depending on the parameters of the planet's deformations—elastic moduli and viscosity coefficients of various media, as well as, the disturbance frequency—makes it possible to solve the complex problem of studying the Earth's internal structure. This line of research is a branch of geophysics. But the problem of the deformable Earth motion relative to its center of mass is a complex task and can combine elements of various fields of science: astrometry, celestial mechanics,

geophysics, the theory of stochastic systems, and many others. And first of all, the methods and approaches used to construct the Earth motion model depend on the goals of scientific research. In that case, if the goal of the problem is modeling in a certain "average" sense that is a development of a model described the motion in question with average observed parameters, then the celestial–mechanical approach seems to be the most rational as the basis for constructing a complex model. Along with this, it is justified from the point of view of practical application to construct a few-parameter mathematical forecast model that allows us to reduce the computational complexity of the algorithmic implementation of the model of the Earth orientation parameters oscillations.

Indeed, for qualitative conclusions about the Earth motion around its center of mass, it will be logically justified to take into account coherent oscillations in various deformable (visco-elastic and liquid) Earth's media. For example, Fig. 15.3a shows the observed oscillations of the gravitational acceleration normal component δg on an SG gravimeter in Membach (Belgium), whose position is marked on the static

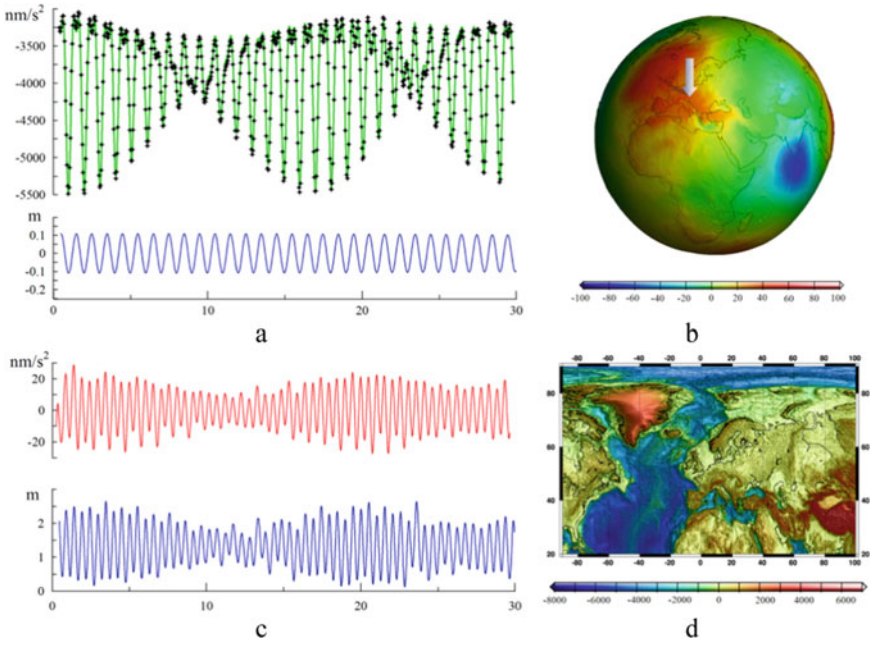


Fig. 15.3 Earth motion observation: **a** variations in the gravitational acceleration according to measurements on an SG gravimeter in Membach (discrete points) in comparison with fluctuations in the model of solid-state tides (green line) and diurnal variations in sea level according to PSMSL station near Rorvik, **b** model of the geoid of the GFZ center (the arrow shows the location of the city of Membach), **c** comparison of hydrosphere tidal oscillations in the data of the gravitational acceleration of the Membach city (red line) and sea-level fluctuations in the Rorvik city (blue line), and **d** geoid elevation map for a portion of the Earth's surface according to the model of the GFZ center (the flag marks the location of Rorvik)

geoid of the GFZ model [14]. An example of comparing the δg variations due to solid-state tides with the measurement data shows that the combined tidal variations of the Earth's deformable media determine 98% of the observed oscillations. These coherent fluctuations in geomedia can be identified with the ones of a viscoelastic thin layer of the adopted deformable Earth model. Then the differences between the solution of such a model problem obtained in the first approximation and observed processes will be in the proportionality coefficients. In turn, these coefficients can be identified with a sufficient degree of accuracy from astrometric and geodetic observations and measurements data. This approach allows some generalization in the case of taking into account hydrosphere oscillations. Indeed, if the oceanic oscillations are taken into account, we can assume that the remaining discrepancy in the oscillations of the measured signal δg along with the influence of atmospheric pressure [15] will also be caused by hydrosphere fluctuations from a relatively small (on the scale of the entire Earth's surface) neighborhood. Variations in atmospheric pressure are usually non-stationary and measured directly at the point of observation. The corresponding fluctuations in the gravitational acceleration can be considered proportional to atmospheric pressure [15, 16]: they can be easily filtered out. However, atmospheric fluctuations in the high-frequency range like any tidal variations of the atmosphere are small. Therefore, the remaining 2% of the amplitude of the high-frequency g oscillations will be due to hydrosphere fluctuations. As an example of the correlation between the variations in the gravitational acceleration and local hydrosphere, a comparison is made (Fig. 15.3b) between the sea-level fluctuations at the coastline of Rorvik (Norway) marked on the map without the long-period component and the corresponding component isolated from δg. Also, in Fig. 15.3a, the gravitational accelerations and close to diurnal sea level variations are compared. For example, if the residual between the measurement data and tidal model of the solid-state oscillations of the gravitational acceleration is represented as the sum of the diurnal and semidiurnal variations $\delta g^{\varphi} + \delta g^{2\varphi}$, then it correlates with the variation $\delta h^{\varphi} - \delta h^{2\varphi}$ of the sea level.

15.4 Geophysical Factors in the Model of the Earth Pole Oscillatory Process

It is well-known [1, 17, 18] that the amplitude and phase of the Chandler component of the Earth pole oscillatory process are very sensitive to various perturbing factors including those with irregular properties (oceanic, atmospheric, and possibly others). The magnitude of the amplitude of the steady-state motion is determined by the frequency difference and dissipation coefficient. Therefore, the Chandler component of the Earth pole oscillations should be considered the most sensitive to the irregular impacts. The mechanism of these impacts is naturally related to weak inertia tensor perturbations.

Even the regular tidal potential [16], due to the complexity of the topography of the global ocean floor and contours of the continents coastlines, leads to the development of a random displacement field and occurrence of random fluctuations in tidal processes. These perturbations correspond to weak irregular perturbations of the Earth's inertia tensor components. However, due to the uneven distribution of the global ocean's water masses over the Earth's surface, their manifestation in the centrifugal moments of inertia J_{xz}, J_{yz}, and, therefore, in the coordinates x_p, y_p, are different.

In Fig. 15.4, the amplitude spectrum of the Earth pole coordinates in the axes x, y (left graph) and x', y' (right graph) are shown. The axes x, y correspond to the terrestrial coordinate system ITRS axes [1] (the axis x is located in the Greenwich meridian plane, and axis y is in the plane orthogonal to it). In turn, axes x', y' are obtained by rotating x, y by an angle determined from the fulfillment of the combined condition of the noise's highest level (the level of the spectral power density of the

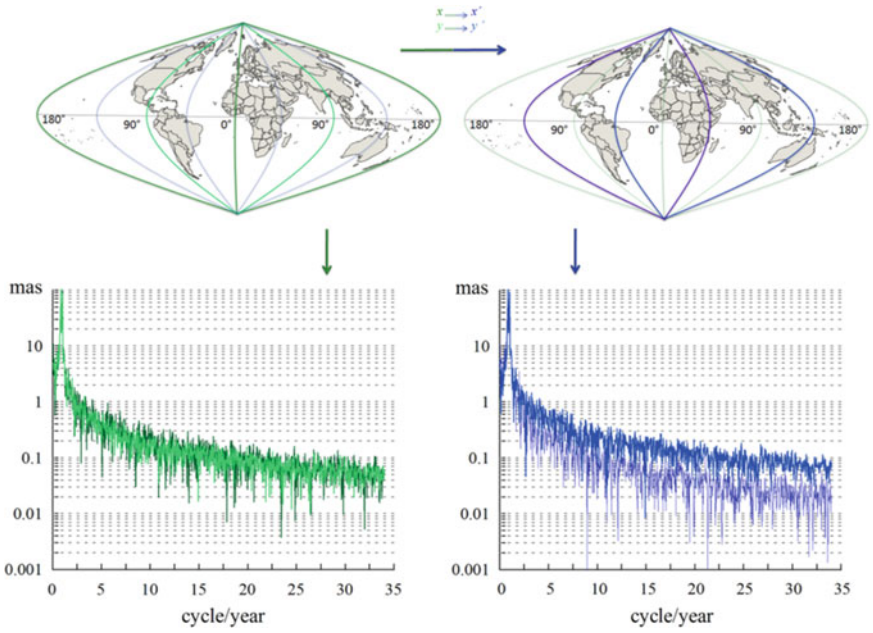


Fig. 15.4 Amplitude spectrum of the Earth pole coordinates. At the bottom left, we see the amplitude spectra of the oscillations of the Earth pole coordinates in the projection on the axis x, y (dark green and light green lines, respectively). At the bottom right, we see the amplitude spectra of oscillations of the Earth pole coordinates in the projection on the axis x', y' (dark blue and blue lines). The upper figure illustrates a relative position of the axes x, y (dark green and light green lines) corresponding to the zero meridian and the 90th meridian of west longitude (top left) and the axes x', y' (dark blue and blue lines, respectively) obtained by turning the first two at an angle of 40° toward the east (top right). The logarithmic scale for amplitudes was used along the ordinate axis of the spectral graphs. The graphs show differences in the harmonics amplitudes of the high-frequency regions along the corresponding axes before and after the rotation

oscillations of the Earth pole coordinates $x'_p,\ y'_p$) in one of the coordinates and the lowest level in the other. The calculations were carried out in a 5° increment.

The resulted graphs show the differences in the harmonics amplitudes of the high-frequency regions. The logarithmic scale for amplitudes was used along the ordinate axis of the graphs. The graphs show that the lowest level of noise (in the frequency range from 5 to 40 cycles per year) is observed at the coordinate x', rotated by an angle of about 40° to the east of Greenwich. The highest level of high-frequency oscillations approximately corresponds to the y' axis, which preserves the direction orthogonal to the x' axis, although the maximum is less explicit than the minimum along the x' axis.

The correspondence of the positions of the axes x', y' to the distribution of water mass over the Earth's surface can be shown visually using simple reasoning. First, we determine the dependence of the total ocean surface ratio to the land surface on longitude. To obtain accurate results, topographic data should be used, followed by their integration over latitude. However, since a high accuracy is not required for a qualitative analysis, we can consider a more original method, which is quite suitable for the purposes of this work. In [19], the results of broadband photometry of the Earth were presented according to the data from the Deep Impact spacecraft operating under the EPOXI mission, and the dependence of the land surface distribution on the longitude was constructed on the basis of light curves. Denote by $k(\theta)$ the share of the ocean surface at longitude θ is defined by Eq. 15.8.

$$k(\theta) = \frac{\text{Ocean Surface at longitude } \theta}{\text{Earth Surface at longitude } \theta} \tag{15.8}$$

Variations in the centrifugal moments of inertia $J_{x'z}$, $J_{y'z}$ have a perturbing effect on the Earth pole oscillatory process. Since the centrifugal moments of inertia characterize the masses distribution relative to the coordinate planes $x'z,\ y'z$, their sensitivity is higher to the motion of the moving medium on the Earth's surface that is located closer to the corresponding plane. Then they will have the greatest sensitivity to tangential displacements in a sector bounded by two meridians and containing a plane with respect to which the moments of inertia are calculated. In this case, the motion of particles on the surface bounded by such a sector occurs in tangential directions, but their latitude is unknown, since the introduced coefficient $k(\theta)$ as an integral value does not depend on latitude and there is no resolution on latitude.

Now we choose two sectors that are symmetrical with respect to the coordinate planes $x'z,\ y'z$ with angles at the vertex $2\theta_0$. When the axes are rotated, the selected sectors will also rotate. If the moving medium is evenly distributed over the surface of the hemisphere (e.g., for $x' > 0$) and is "frozen" on the opposite hemisphere, then the particle motion on the surface bounded by such a sector determine approximately $100 \sin\theta_0\%$ of the variable part for the centrifugal moments of inertia (at $\theta_0 = \pi/2$ the sector becomes a hemisphere). For an unevenly distributed medium, this value can differ and the smaller the angle $\theta_0 \in (0, \pi/2]$, the larger the difference. But on the other hand, the larger the angle θ_0 (i.e., the larger the area of the surface under consideration), the greater the uncertainty of the correspondence between the ocean

distribution and the total coefficient $k(\theta)$ in the sector after integration over longitude is. Therefore, the choice of the sector's angle (basically, the choice of the integration region of the coefficient $k(\theta)$ in longitude) is a compromise between the maximum sensitivity of centrifugal moments of inertia to the particles motion along a surface that is limited by the sector and the minimum area of this surface in order to reduce the uncertainty error. Since the share of the surface area limited by a sector is $2\theta_0/\pi$, θ_0 is determined from the condition that the function $\pi \sin\theta_0 - 2\ \theta_0$ is maximum on the $0 < \theta_0 \leq \pi/2$ interval. Under the condition $\theta_0 \approx 0.9$, the motion of the particles in this sector determines approximately 80% variations of the tesseral harmonic of the geopotential. However, let us choose a slightly larger value of the angle and, in the following formulas, put for illustration purposes $\theta_0 = \pi/3$, although this will not fundamentally affect the estimates of average values.

Let us determine the share of the ocean surface area limited by one selected sector. Since the Earth's surface area limited by a sector is a constant and does not depend on the Earth rotation, the share of the ocean's surface area in the sector with an vertex angle as $(\theta - \pi/3, \theta + \pi/3)$ is proportional to the average coefficient $k(\theta)$:

$$\overline{k}(\theta) = \langle k(\theta)\rangle_{2\pi/3} = \frac{3}{2\pi}\int\limits_{\theta-\pi/3}^{\theta+\pi/3} k(\theta)d\theta. \tag{15.9}$$

The centrifugal moments of inertia $J_{x'z}$, $J_{y'z}$ are most sensitive to the motion of the moving medium if the ocean distributions in the selected sector and in the sector symmetrical to it are significantly different. This condition can be replaced in a non-strict sense by the integral condition $\overline{k}(\theta) - \overline{k}(\theta + \pi) \neq 0$. In the strict sense, this condition does not appear directly from Eq. 15.9 and thus is taken as an assumption. If the location of the axes x', y' meets this condition, then the assumption will be valid.

In order to establish a correspondence, a function is defined

$$f(\theta) = \left[\overline{k}(\theta) - \overline{k}(\theta + \pi)\right]^2 \tag{15.10}$$

that when $\overline{k}(\theta) = \overline{k}(\theta + \pi)$ takes the minimum value, i.e., with equal share of the ocean surface in two opposite sectors, and the maxima corresponds to the extrema of the function $\overline{k}(\theta) - \overline{k}(\theta + \pi)$, when the share of the ocean surface in two opposite sectors are most different.

The correspondence between the location of the axes x', y' and the distribution of the ocean over the Earth's surface is shown in Fig. 15.5. It can be seen that the directions of the axes approximately correspond to the extrema of the function $f(\theta)$.

That is, in the approximately orthogonal direction to the axis x' one can assume a minimum of the amplitude of high-frequency perturbations due to less asymmetry in the ocean distribution, which, according to the results of processing the pole motion data, leads to high-frequency oscillations along the coordinate x' with lower intensity. Similarly, with respect to the coordinate y', the oscillations with a higher

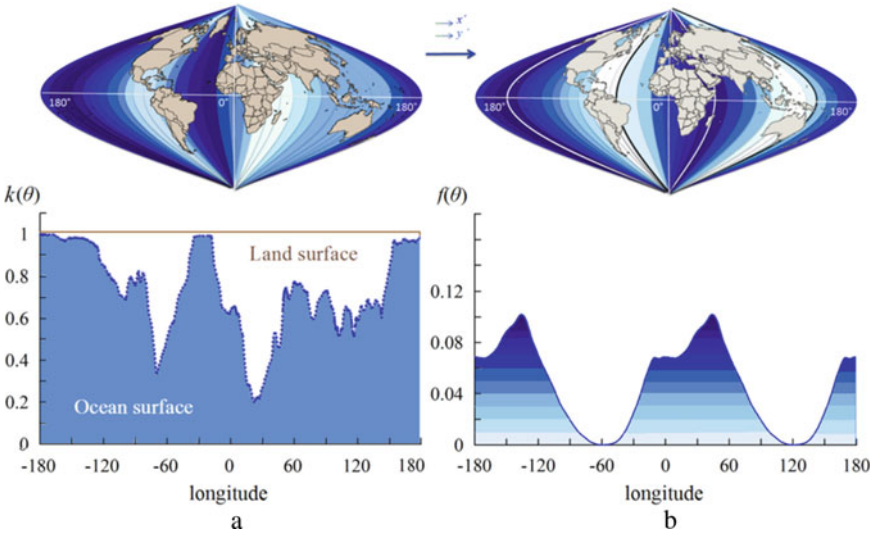


Fig. 15.5 Real and estimated distributions: **a** distribution of the ocean (blue color) and land (white color) on the Earth's surface depending on longitude (top) and share of the ocean surface at longitude θ (bottom), **b** distribution of the function $f(\theta)$ plotted on the Earth's map and the location of the axes x', y' corresponding to Fig. 15.4 (top) and dependence of the function $f(\theta)$ and the correspondence of its extrema to the location of the axes x', y' (bottom)

intensity are observed due to the greater asymmetry of the ocean distribution along the axis x'. Although it does not follow directly from the maximum in the axis y' and the minimum in the axis x' of the short-period pole oscillations that the intensity of the high-frequency perturbation in the projection onto the axis x' exceeds the intensity of the perturbation in the projection onto the axes y', and not, for example, vice versa. Moreover, the maximum and minimum amplitudes of high-frequency perturbations can be achieved also while projecting on non-orthogonal axes. But from the analysis of the calculated total geodetic perturbations and separately the ocean perturbations in the projection on the axes x', y', it can be established that the highest intensity of high-frequency perturbations is observed in the projection on the axis of approximately 15° of the east longitude and the lowest intensity is about 75° of the west longitude. These directions differ from the directions of the axes x', y' in the projection onto which the extrema of the amplitudes of the short-period Earth pole oscillations are observed, but with the same error correspond to the extrema of the function $f(\theta)$. Of course, for more accurate conclusions, it is important not only to estimate the asymmetry in the distribution of various media over the surface, but also to quantify the distributions, as well as, the latitude distribution. However, the calculations performed allow us to draw some conclusions.

Thus, the orientation of the vector of complete geodesic perturbations including the influence of the atmosphere and the ocean corresponds to the distribution of the ocean over the Earth's surface in the sense considered above. Consequently,

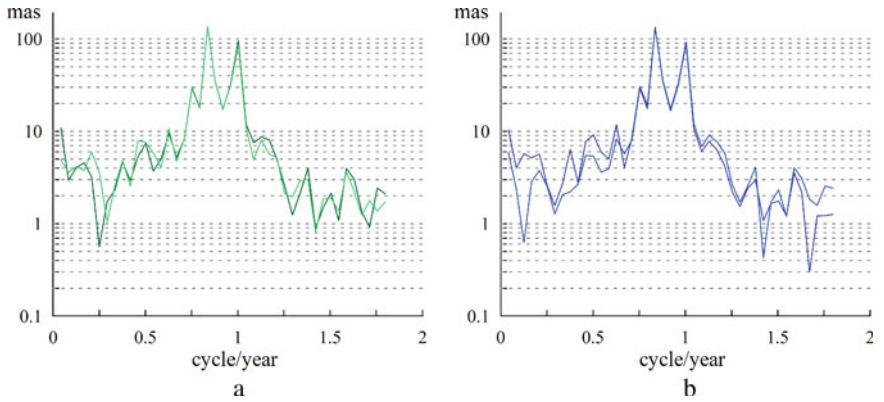


Fig. 15.6 Amplitude spectra of the oscillations of the Earth pole coordinates in the projection: **a** on the axis x, y (dark green and light green lines, respectively), **b** on the axis x', y' (dark blue and blue lines, respectively)

geophysical perturbations are some consistent oscillations of moving media and can be considered together as a combination.

The directions of the axes x', y' found from the condition of maximum and minimum intensities of perturbations approximately correspond to the distribution of the ocean over the Earth's surface, not only for high-frequency oscillations, but also for oscillations from any not too short frequency interval. That is, the correspondence can be shown for the entire spectrum of oscillations with the only caveat that for perturbations with frequencies below the Chandler frequency the arrangement of the axes x', y', it will change by 90°. This means that the maximum amplitude of the pole oscillations at a frequency above the Chandler's one will be observed along the axis y', and the maximum amplitude at a frequency below the Chandler's one will be observed along the axis x' (Fig. 15.6). This circumstance is also due to the correspondence of the phases with the fluctuations spectrum onto which the perturbations are decomposed. And this also indicates the consistency of perturbations of various physical nature.

15.5 The Role of Astronomical Factors in the Perturbed Earth Pole Motion

It is known [20] that coherent oscillations in various media can appear in the geophysical processes on a planetary scale. A number of large-scale phenomena of the atmosphere and the ocean, the global seismic activity of the Earth have signs of common oscillatory processes also inherent in the Earth rotational motion [21–23]. And the Chandler wobble is no exception. However, the process of developing such oscillations has not been sufficiently studied to this moment. For example, the variations

in the main components parameters of the Earth pole oscillations may have more global causes than it is assumed, and the process of their excitation is caused not only by fluctuations of geophysical media of a stochastic nature. More precisely, these oscillations can be non-stationary, but be of a natural nature, and not stochastic. From the result of processing data on the Earth pole motion, it appears [18] that the oscillations of the Earth's moving media in the spectral range of the Chandler and annual harmonics turn out to be ordered in some way. For example, in the observed Earth pole motion, it is possible to establish the presence of an in-phase oscillatory process with a precession of the lunar orbit [9, 18].

The spatial motion of the lunar orbit consists of a series of rotations around intersecting axes. They lead to the cyclical motion of its nodes and perigee [24]. In addition, the derivatives of the orbit parameters are nonzero and are varying values, being the subject to small variations. A result of the lunar orbit precession and of the associated cyclic change in the longitude of the ascending node with a period of 18.61 years is a change in the orbit plane inclination to the Earth's equator. The inclination of the lunar orbit to the Earth's equator varies from 18.3° to 28.58°. In this case, the point of intersection of the lunar orbit circle with the equator oscillates along the equator near its average position, which coincides with the point of the vernal equinox. Unlike the node (the intersection point of the lunar orbit circles and the ecliptic in the celestial sphere), which makes a complete revolution, the intersection point of the orbit and the equator oscillates in the range from –13.2° to 13.2°.

In [25], it was shown that one can find a transformation of the Earth pole coordinates, illustrating in-phase nature of its Earth pole oscillatory process and the lunar orbit precession. Namely, the oscillatory motion of the pole minus the Chandler (or annual depending on the amplitudes values of the Chandler and annual harmonics) and six-year cycles occurs in-phase with oscillations along the equator of the intersection point of the lunar orbit and the equator. This feature requires a more detailed analysis and study of the causes of such fluctuations. In particular, it is of interest to establish the contribution of geophysical (atmospheric and oceanic) disturbances to these oscillations.

As a result of the numerical solution of the differential equations of the Earth pole motion, the trajectories of the pole are obtained for various perturbations. The perturbing functions were tabulated according to the IERS published data. For example, in Fig. 15.7, it is shown a comparison between the fluctuations in the calculated motion of the Earth pole taking into account the combined perturbations from the atmosphere and the ocean and the fluctuations of its observed motion.

To isolate the oscillatory process with a frequency of 0.05373 cycle/year from the calculated and observed pole oscillations, the procedure proposed in [25] was applied. Using transformations of the Earth pole coordinates, the essence of which is the elimination of two cycles—with the Chandler and six-year periods, it is possible to obtain a pole oscillation in-phase with the precession of the lunar orbit. In Fig. 15.8a, a comparison is shown between the variations of the polar angle φ isolated from the observed Earth pole trajectory, its approximation by a two-frequency model with constant coefficients, and the calculated Earth pole trajectory taking into account atmospheric and oceanic perturbations. In the lower graph of Fig. 15.8, a graph of

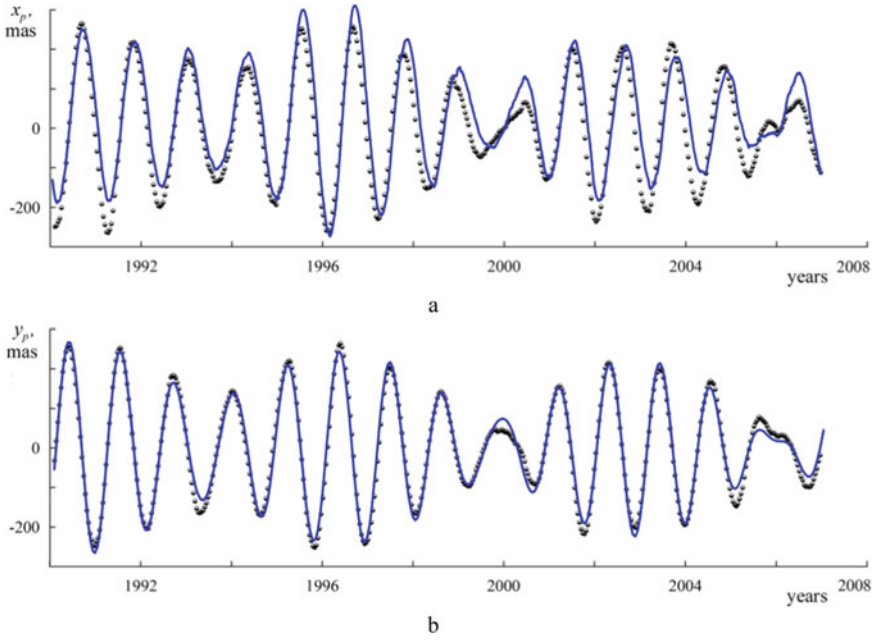


Fig. 15.7 Earth pole oscillations according to the IERS observations and measurements (discrete data) in comparison with the calculated oscillations caused by perturbations of: **a** atmosphere and **b** ocean

oscillations of the angle of deviation δ along the equator of the point of intersection of the equator with the lunar orbit is constructed. The main harmonic with the precession frequency of the lunar orbit for the observed pole motion is shown by the red line, and the blue dots are for the calculated motion taking into account geophysical perturbations. The oscillations caused by geophysical perturbations have much smaller amplitude and shifted phase, which indicate more complex physical nature of these oscillations and the incompleteness of the disturbances taken into account.

15.6 Conclusions

Variations in the main components parameters of the Earth pole motion are due to the effect of the combined nature. The considered main geophysical perturbations are apparently part of the coherent oscillations of various media. Not more than 50% of the energy of the considered oscillatory process is due to perturbations of the atmosphere and the ocean. Since the impact of other geophysical fluids on the Earth pole motion is much smaller, this process should be more global in nature and such variations in the Earth's environment can be affected. Then, their excitation

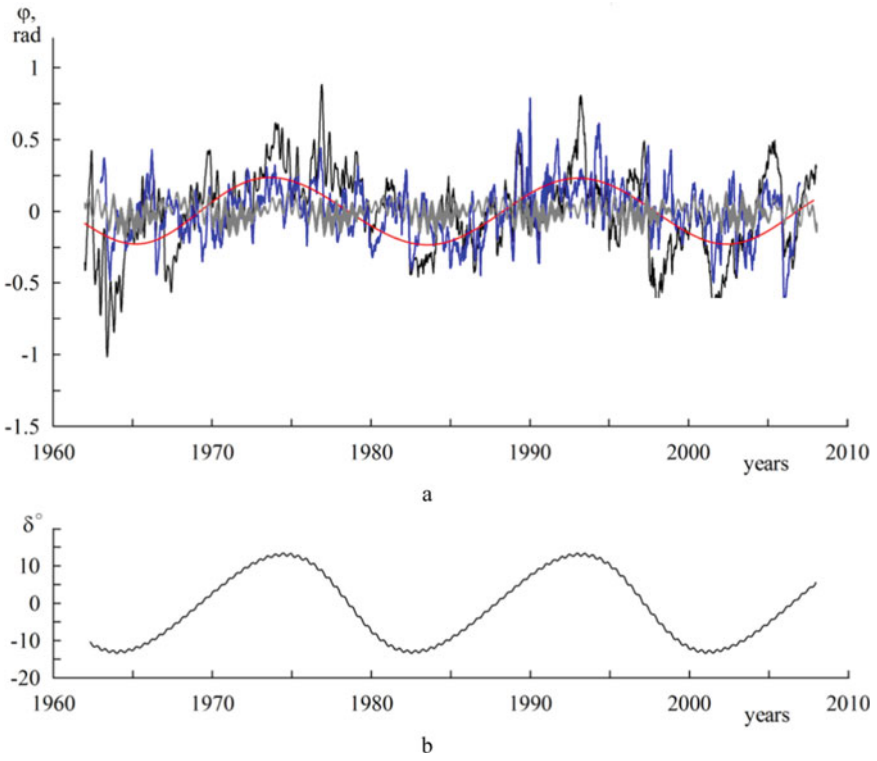


Fig. 15.8 Estimates of the polar angle variations: **a** comparison between the variations of the polar angle φ isolated from the observed Earth pole trajectory (black line), its approximation (red line), variations of the polar angle φ using a two-frequency model with constant coefficients (gray line), and the calculated Earth pole trajectory taking into account the atmospheric and oceanic perturbations (blue line) and **b** a graph of oscillations of the deviation angle δ along the equator of the point of intersection of the equator with the lunar orbit

in geomedia can be caused not so much by internal perturbations as by external disturbances for the Earth.

Acknowledgements This work was carried out within the basic part of the state task of the Ministry of Education and Science of the Russian Federation (project no. 721).

References

1. International Earth rotation and reference systems service—IERS annual reports, https://www.iers.org. Last accessed 9 Sept 2020
2. Markov, Y.G., Mikhaylov, M.V., Lar'kov, I.I., Rozhkov, S.N., Krylov, S.S., Perepelkin, V.V., Pochukaev, V.N.: Fundamental components of the parameters of the Earth's rotation in forming

high-precision satellite navigation. Cosmic Res. **53**, 143–154 (2015)
3. Markov, Y.G., Mikhailov, M.V., Perepelkin, V.V., Pochukaev, V.N., Rozhkov, S.N., Semenov, A.S.: Analysis of the effect of various disturbing factors on high-precision forecasts of spacecraft orbits. Cosmic Res. **54**, 155–163 (2016)
4. Petukhov, V.G.: Application of the angular independent variable and its regularizing transformation in the problems of optimizing low-thrust trajectories. Cosmic Res. **57**, 351–363 (2019)
5. Ivanyukhin, A.V., Petukhov, V.G.: Low-energy sub-optimal low-thrust trajectories to libration points and halo-orbits. Cosmic Res. **57**, 378–388 (2019)
6. Starchenko, A.E.: Trajectory optimization of a low-thrust geostationary orbit insertion for total ionizing dose decrease. Cosmic Res. **57**, 289–300 (2019)
7. Bizouard, C., Remus, F., Lambert, S., Seoane, L., Gambis, D.: The Earth's variable Chandler wobble. Astron. Astrophys. **526**, A106.1–A106.4 (2011)
8. Munk, W.H., MacDonald, G.J.F.: The rotation of the Earth. Cambridge University Press, New York (1961)
9. Markov, Y.G., Perepelkin, V.V., Filippova, A.S.: Analysis of the perturbed Chandler wobble of the Earth pole. Dokl. Phys. **62**(6), 318–322 (2017)
10. Krylov, S.S., Perepelkin, V.V., Filippova, A.S.: Long-period lunar perturbations in Earth pole oscillatory process: Theory and observations. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Theory and Practice of Computational Mechanics. SIST, vol. 173, pp. 315–331. Springer, Singapore (2020)
11. Zlenko, A.A.: A celestial-mechanical model for the tidal evolution of the Earth-Moon system treated as a double planet Astron. Rep. **59**(1), 72–87 (2015)
12. Zlenko, A.A.: The perturbing potential and the torques in one three-body problem. J. Phys. Conf. Ser. **1301**, 012022.1–012022.12 (2019)
13. Akulenko, L.D., Markov, Y.G., Rykhlova, L.V.: Motion of the Earth's poles under the action of gravitational tides in the deformable-Earth model. Dokl. Phys. **46**(4), 261–263 (2001)
14. Information System and Data Center for geoscientific data, https://isdc.gfz-potsdam.de, last accessed 2020/09/09.
15. Guochang, Xu.: Sciences of Geodesy—I: Advances and Future Directions. Springer, Berlin, Heidelberg (2010)
16. Schubert, G.: Treatise on Geophysics, vol. 3. Elsevier, Geodesy (2007)
17. Kumakshev, S.A.: Gravitational-tidal model of oscillations of Earth's poles. Mech. Solids **53**(2), 159–163 (2018)
18. Kumakshev, S.A.: Model of oscillations of Earth's poles based on gravitational tides. In: Karev, V., Klimov, D., Pokazeev, K. (eds.) Physical and Mathematical Modeling of Earth and Environment Processes. PMMEEP, 157–163 (2017)
19. Cowan, N.B., Agol, E., Meadows, V.S., Robinson, T., Livengood, T.A., Deming, D., Lisse, C.M., A'Hearn, M.F., Wellnitz, D.D., Seager, S., Charbonneau, D.: Alien maps of an ocean-bearing world. Astrophys. J. **700**(2), 915–923 (2009)
20. Sidorenkov, N.S.: The interaction between Earth's rotation and geophysical processes. Wiley-VCH Verlag GmbH and Co, KGaA (2009)
21. Sidorenkov, N.S.: Synchronization of terrestrial processes with frequencies of the Earth–Moon–Sun system. AApTr **30**(2), 249–260 (2017)
22. Sidorenkov, N.S.: The Chandler wobble of the poles and its amplitude. In: Proceedings of the "Journees 2014 Systemes de reference spatio-temporels", Pulkovo observatory, Russia, pp. 195–197 (2014)
23. Sidorenkov, N.S.: Celestial mechanical causes of weather and climate change. Izv. Atmos. Ocean. Phys. **52**, 667–682 (2016)
24. Smart, W.M.: Celestial Mechanics. Longmans, Green (1953)
25. Perepelkin, V.V., Rykhlova, L.V., Filippova, A.S.: Long-period variations in oscillations of the Earth's pole due to lunar perturbations. Astron. Rep. **63**(3), 238–247 (2019)

Chapter 16
Application of Multi-agent Optimization Methods Based on the Use of Linear Regulators and Interpolation Search for a Single Class of Optimal Deterministic Control Systems

Andrei V. Panteleev and Maria Magdalina S. Karane

Abstract Two new multi-agent algorithms for controlling one class of deterministic systems are proposed: the hybrid multi-agent method of interpolation search and multi-agent method based on the use of linear regulators of agent movement control. Detailed descriptions of the strategies of these methods are given, and step-by-step algorithms for each multi-agent method are described. Since multi-agent algorithms are used to find optimal control of dynamic systems, step-by-step algorithms for finding optimal open-loop control using multi-agent methods are also given. Two approaches to the search for optimal open-loop control are considered: when control is sought in relay form with a certain number of switches and when control is sought in the form of an expansion in a system of basis functions. In this chapter, cosine curves were considered as basis functions. Based on the above algorithms, software has been formed that allows finding the optimal open-loop control. Recommended parameters are given for each multi-agent algorithm. To study the effectiveness of the above algorithms, a specially selected set of test problems for finding the optimal open-loop control is solved, where the model of the control object is described by an ordinary differential equation linear in bounded control. During the study, it was shown that the described algorithms successfully cope with the task and can find a solution close to the exact one.

A. V. Panteleev · M. M. S. Karane (✉)
Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, Moscow 125993, Russian Federation
e-mail: mmarselina@mail.ru

A. V. Panteleev
e-mail: avpanteleev@inbox.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_16

16.1 Introduction

The scope of multi-agent algorithms [1] is quite wide, and at present, such algorithms are more and more often used to solve various kinds of optimization problems. This is due to the fact that multi-agent algorithms are in no way inferior to existing classical methods, and even vice versa quite often surpass them. They make it possible to solve problems of greater dimension much more successfully or if restrictions are imposed on the system. The advantage of multi-agent algorithms also lies in the fact that it is not necessary to have information about the behavior of a function or its properties. Multi-agent methods are used in many fields, such as the theory of optimal control [2–5] for optimizing a criterion or in machine learning for tuning and training neural networks [6].

The principle of operation of multi-agent algorithms consists in the formation of a group of agents on the solution search set, and depending on the specific algorithm, a set of actions is carried out on the agents that lead group of agents to answer the task.

The purpose of this work is to develop a multi-agent algorithms and their application in order to find the optimal open-loop control. This requires a formation of more general algorithm, which will include multi-agent algorithms. Another purpose is to find the optimal open-loop control in two ways: by decomposing the control into a system of basis functions and representing the control in relay form with a certain set of switching points.

The novelty of this approach is the use of multi-agent algorithms to search for optimal program control. Each multi-agent algorithm is based on new ideas for finding the optimal solution. The novelty of the hybrid multi-agent method of interpolation search [7] is the use of interpolation curves, which allows to adapt the locally changing structure of the level surfaces of the objective function. In the multi-agent method based on the use of linear regulators of agent movement control, four types of optimal program control with full feedback on the state vector are searched at the stages of the algorithm. For each control, its own criterion is optimized.

In addition, before applying multi-agent algorithms, one should investigate their effectiveness on a standard set of test functions [8, 9] of two variables in order to identify the most suitable ranges of parameter values. Using them, it is possible to solve applied problems with great success. In [7], for the hybrid multi-agent method of interpolation search, a detailed analysis of the efficiency is given, and the best algorithm parameters are established.

The chapter is organized as follows. Section 16.2 provides a description of multi-agent methods. Application of multi-agent methods for optimal open-loop control problems is given in Sect. 16.3. Section 16.4 concludes the chapter.

16.2 Description of Multi-agent Methods

Hereinafter, Sect. 16.2.1 discusses an optimization problem. Hybrid multi-agent optimization method of interpolation search is presented in Sect. 16.2.2. Multi-agent optimization algorithm using linear regulators for agents' motion control is developed in Sect. 16.2.3.

16.2.1 Optimization Problem

It is given the objective function $f(x) = f(x_1, x_2, \ldots, x_n)$ defined on the set of admissible solutions $D \subseteq R^n$. It is required to find the constrained global maximum of a function $f(x)$ on set D, i.e., such a point $x^* \in D$, that

$$f(x^*) = \max_{x \in D} f(x), \tag{16.1}$$

where $x = (x_1, x_2, \ldots, x_n)^T$, $D = \{x | x_i \in [a_i, b_i],\ i = 1, 2, \ldots, n\}$.

The task of finding the minimum of a function $f(x)$ is replaced by the task of finding the maximum by replacing the sign before the function with the opposite: $f(x^*) = \min_{x \in D} f(x) = -\max_{x \in D}[-f(x)]$. Function $f(x)$ can be multiextremal, so the required solution in the general case is not unique.

16.2.2 Hybrid Multi-agent Optimization Method of Interpolation Search

Solution search strategy. The search strategy includes interpolation search, which uses several points of the current population and reduces the task of finding new solutions to the problems of one-dimensional parametric maximization, swarm intelligence method to maximize the objective function value along the interpolation curve, and self-organizing migrating algorithm [7].

The considered objective function $f(x)$ is called the fitness function, and the vector of parameters $x = (x_1, x_2, \ldots, x_n)^T$ of the objective function is an individual. Each vector $x = (x_1, x_2, \ldots, x_n)^T \in D$ is a possible solution of an optimization problem. The smaller the value of the objective function $f(x)$, the more the individual x is adapted, i.e., suitable as a solution.

In solving problem (Eq. 16.1), finite sets $I = \{x^j = (x_1^j, x_2^j, \ldots, x_n^j)^T,\ j = 1, 2, \ldots, NP\} \subset D$ of possible solutions are used. These solutions are called populations, where x^j is the individual with the number j and NP is the size of the population.

The hybrid multi-agent method of interpolation search imitates the evolution of the initial population $I_0 = \{x^j,\ j = 1, 2, \ldots, NP | x^j = (x_1^j, x_2^j, \ldots, x_n^j)^T \in D\}$ and is an iterative process that explores the set D.

The procedure of finding a solution begins with the generation of the initial population of individuals x^j ($j = \overline{1, NP}$) on the set D using a uniform distribution. The first phase of the search is interpolation search. The construction of different interpolation polynomials allows to adapt a locally changing structure of the level surfaces of the objective function. A different role of leading points is also used to realize frontal search or deep search of an admissible solution set, thereby, providing additional flexibility of the search strategy. The interpolation polynomials type choice is optional. Thus, the choice of the interpolation polynomial type and the points, by which it is formed, implements two types of search: exploration and exploitation.

To implement it, four members P_1, P_2, P_3, P_4 in the population are selected. Among them $P_1 = x^{(1)}$ is a leader, and P_2, P_3, P_4 are random members of the population. All four points are different. The Bezier curve is used to process them. It passes inside the convex hull formed by the selected four points. As $t = 0$, curve passes through P_1, and as $t = 1$, curve passes through point P_4. Next, we find the solution to the parametric optimization problem

$$x^{\text{Bezier4}} = \arg \max_{t \in [0,1]} f[(1-t)^3 P_1 + 3(1-t)^2 t P_2 + 3(1-t)t^2 P_3 + t^3 P_4], \quad (16.2)$$

and a new member x^{Bezier4} is added to the population.

B-spline curve is used to explore new areas. The curve is formed by four random members of the population P_1, P_2, P_3, P_4 that are different

$$x^B = \arg \max_{t \in [0,1]} f\left[\frac{1}{2}\left[-t(1-t)^2 P_1 + (2 - 5t^2 + 3t^3) P_2 + t(1 + 4t - 3t^2) P_3 - t^2(1-t) P_4\right]\right]. \quad (16.3)$$

As a rule, the curve does not pass through any point; it is in the convex hull generated by four vertices. As a result, one more new member x^B is added to the population.

The second phase of the search is the migration of the population. The leader is selected in the population (the best solution) $x^{(1)}$. All other members of the population $x^{(j)}$, $j = 2, ..., NP$ move toward the leader making $nstep$ discrete steps, and half of these steps being done to the leader, and then as many more steps are taken in the same direction. The new position of a member of the population is determined by the best decision reached during this search. The direction of the search is given by a vector *PRTVector*, whose coordinates are zero or one. If the coordinate is zero, the search for this coordinate is not conducted, and if it is one, then it is performed. Thus, the solution to the problem is sought in all coordinates that are simultaneously equal to one. The position of the leader in the migration process does not change.

The third phase is the frontal search, which serves to clarify the final solution of the problem. It uses interpolation curves. Information about the position of the first three or four leaders among the members of a population is used to form curves.

Among the members of the population $x^{(1)}, \ldots, x^{(NP)}$ located in ascending order of the value of the fitness function, three leaders $P_1 = x^{(1)}$, $P_3 = x^{(2)}$, $P_2 = x^{(3)}$ are selected, according to which the Bezier curve is formed (as $t = 0$ it passes through point P_1, and as $t = 1$ it passes through point P_3). Next we find the solution to the parametric optimization problem

$$x^{\text{Bezier3}} = \arg \max_{t \in [0,1]} f[(1-t)^2 P_1 + 2(1-t)t P_2 + t^2 P_3], \tag{16.4}$$

and a new member x^{Bezier3} is added to the population.

Four points $P_1 = x^{(3)}$, $P_2 = x^{(1)}$, $P_3 = x^{(2)}$, $P_4 = x^{(4)}$ are selected to continue the search. The Catmull–Rom interpolation curve passes through the two best point (as $t = 0$ it passes through point P_2, and as $t = 1$ it passes through point P_3). Next we find the solution to the parametric optimization problem

$$x^{CR} = \arg \max_{t \in [0,1]} f\left[\frac{1}{2}\left[-t(1-t)^2 P_1 + (2 - 5t^2 + 3t^3) P_2 + t(1 + 4t - 3t^2) P_3 - t^2(1-t) P_4\right]\right] \tag{16.5}$$

and a new member x^{CR} is added to the population.

The Bezier curve formed by the four population leaders $P_1 = x^{(1)}$, $P_4 = x^{(2)}$, $P_2 = x^{(3)}$, $P_3 = x^{(4)}$ is used for a similar search (as $t = 0$ it passes through point P_1, and as $t = 1$ it passes through point P_4). Next we find the solution to the parametric optimization problem

$$x^{\text{Bezier4}} = \arg \max_{t \in [0,1]} f[(1-t)^3 P_1 + 3(1-t)^2 t P_2 + 3(1-t)t^2 P_3 + t^3 P_4], \tag{16.6}$$

and a new member x^{Bezier4} is added to the population.

B-spline curve, which is formed by the four leaders of the population $P_1 = x^{(1)}$, $P_4 = x^{(2)}$, $P_2 = x^{(3)}$, $P_3 = x^{(4)}$, can be used for frontal search. The curve does not pass through any selected point, but it is in the convex hull generated by these vertices. Next we find the solution to the parametric optimization problem

$$x^{B} = \arg \max_{t \in [0,1]} f\left[\frac{1}{2}\left[-t(1-t)^2 P_1 + (2 - 5t^2 + 3t^3) P_2 + t(1 + 4t - 3t^2) P_3 - t^2(1-t) P_4\right]\right], \tag{16.7}$$

and a new member x^{B} is added to the population.

The procedure of maximization of the objective function value along the interpolation curves.

The first and third phases require one-dimensional maximization of parametric curves. The swarm intelligence method called as Krill Herd [10, 11] is used for maximization, but it was possible to use classical algorithms for one-dimensional maximization, for example, the dichotomy method or the golden-section search. Krill Herd method is based on the results of the krill packs behavior analysis, resembling shrimps. Their positions change under the influence of three factors: the presence of other members of the population, need to search for food, and random walks. Usually the movement of krill population is determined by two goals: the increase in the density of krill and attainment of food. At the beginning of the process, a population NP' is generated from individuals on interval $t \in [0, 1]$ using a uniform distribution. It is assumed that the motion of the j th member of the population occurs according to Eq. 16.8, where x^j is the position, V^j is the speed, which consists of three terms.

$$\frac{\mathrm{d}x^j}{\mathrm{d}t} = V^j \tag{16.8}$$

The first term is determined by the influence of neighbors (members of the population that belong to certain neighborhood of jth element of a certain radius), the best element in the entire population, and information about its former speed. The second term is determined by the movement toward the food source (the "center of mass" of the population is taken for it), information about the former speed in search of food, and the memory of its best result for all the iterations. The third term imitates the random walks of the individual, which decrease with increasing number of iterations. To revive the search process, the cross and mutation operations are used in other evolutionary methods, as well as, the method of differential evolution are applied. The procedure for finding the maximum of the interpolation curve ends when the specified number of iterations is reached. And as an answer from the last krill population, the individual that corresponds to the smallest value of the parametric curve f is selected, and a new member x^j is added to the original population.

The hybrid multi-agent method of interpolation search finishes work after the specified number of iterations is passed. As an approximate solution to the problem, an individual from the last population with the greatest value of the objective function is selected.

Solution search algorithm

Step 1 Set the method parameters: NP is the number of members in the population, M_1 is the number of points obtained using the Bezier curves, M_2 is the number of points obtained using B-spline curves, PRT is the parameter that determines the search activity by coordinate, $n$step is the number of possible positions of the population members, b_2 is the number of individuals (worst) to reduce population, and $I_{\max}$ is the maximum number of iterations. Let $I = 1$ (iteration count).

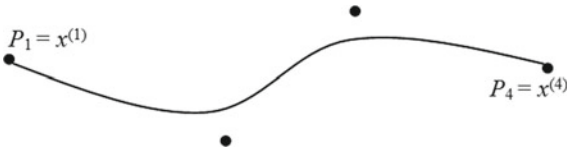


Fig. 16.1 Four point of the Bezier curve

Step 2 Generate the initial population on a set D using the uniform distribution law: $x^1, \ldots, x^{NP}$. Calculate the values of the objective function: $f(x^1), \ldots, f(x^{NP})$.

Step 3 Order the population consisting of NP individuals by the value of the objective function.

Step 4 Execute the interpolation search.

Step 4.1 Perform this step M_1 times. Select $P_1 = x^{(1)}$ (the best) in the current population I_0, and as P_2, P_3, P_4 are three different random members of the population and different from $x^{(1)}$ (Fig. 16.1) find a solution to the problem of parametric optimization provided by Eq. 16.9.

$$x^{\text{Bezier4},j} = \arg\max_{t\in[0,1]} f[(1-t)^3 P_1 + 3(1-t)^2 t P_2 + 3(1-t)t^2 P_3 + t^3 P_4], \quad j = 1, \ldots, M_1 \tag{16.9}$$

Step 4.2 Perform this step M_2 times. Select four different members of population P_1, P_2, P_3, P_4 in the current population I_0 (Fig. 16.2) and find a solution to the problem of parametric optimization provided by Eq. 16.10.

$$x^{B,j} = \arg\max_{t\in[0,1]} f\left[\frac{1}{2}\left[-t(1-t)^2 P_1 + \left(2 - 5t^2 + 3t^3\right) P_2 + t\left(1 + 4t - 3t^2\right) P_3 - t^2(1-t) P_4\right]\right] j = 1, \ldots, M_2 \tag{16.10}$$

Step 4.3 Reduce the population. Place individuals of the current population, replenished with $M_1 + M_2$ new members increasing the value of the fitness function. Leave only NP best.

Step 5 Execute migration of population.

Step 5.1 For any $x^{(j)}$, $j = 2, \ldots, NP$:

Step 5.1.1 Generate $PRTVector^{(j)}$ with coordinates using Eq. 16.11.

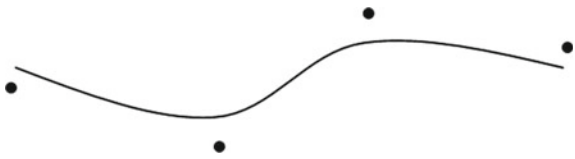


Fig. 16.2 Four point of B-spline

$$PRTVector_i^{(j)} = \left\{ \begin{array}{l} 1 \text{ if } \text{rand}_i < PRT \\ 0 \text{ else} \end{array} \right\} \quad \text{rand}_i = U[0,1] \quad (16.11)$$

Step 5.1.2 Consistently find the probable positions of population members using Eq. 16.12, where $\otimes$ is the component product of vectors (by Hadamard).

$$x^{(j),m} = x^{(j)} + \frac{x^{(1)} - x^{(j)}}{\left[\frac{nstep}{2}\right]} m \otimes PRTVector_i^{(j)} \quad m = 0, 1, \ldots, nstep \quad (16.12)$$

Step 5.1.3 Find the best position of population members during migration using Eq. 16.13.

$$x^{(j),\text{new}} = \arg \max_{m=0,1,\ldots,nstep} f(x^{(j),m}) \quad j = 2, \ldots, NP \quad x^{(1),\text{new}} = x^{(1)} \quad (16.13)$$

Step 5.2 Place new members of the population after migration in ascending order of fitness function value.

Step 6 Execute frontal search.

Step 6.1 Select $P_1 = x^{(1)}$, $P_3 = x^{(2)}$, $P_2 = x^{(3)}$ in the current population (Fig. 16.3) and solve the problem

$$x^{\text{Bezier3}} = \arg \max_{t \in [0,1]} f[(1-t)^2 P_1 + 2(1-t)t P_2 + t^2 P_3]. \quad (16.14)$$

Step 6.2 Select $P_1 = x^{(3)}$, $P_2 = x^{(1)}$, $P_3 = x^{(2)}$, $P_4 = x^{(4)}$ in the current population (Fig. 16.4) and solve the problem

$$x^{CR} = \arg \max_{t \in [0,1]} f\left[\left[\frac{1}{2}[-t(1-t)^2 P_1 + (2 - 5t^2 + 3t^3) P_2 + t(1 + 4t - 3t^2) P_3 - t^2(1-t) P_4]\right]\right]. \quad (16.15)$$

Step 6.3 Select $P_1 = x^{(1)}$, $P_4 = x^{(2)}$, $P_2 = x^{(3)}$, $P_3 = x^{(4)}$ in the current population I_0 (Fig. 16.5) and solve the problem

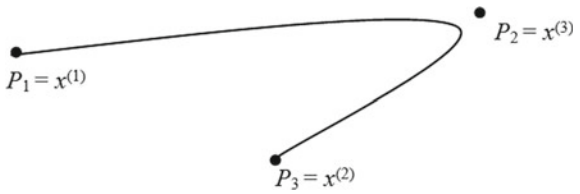


Fig. 16.3 Three point of the Bezier curve

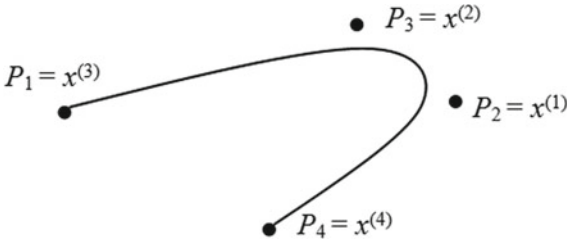


Fig. 16.4 Four point of the Catmull–Rom curve

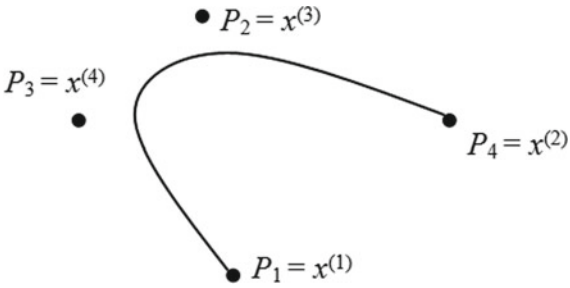


Fig. 16.5 Four point of the Bezier curve

$$x^{\text{Bezier4}} = \arg\max_{t\in[0,1]} f\left[(1-t)^3 P_1 + 3(1-t)^2 t P_2 + 3(1-t)t^2 P_3 + t^3 P_4\right]. \tag{16.16}$$

Step 6.4 Select $P_1 = x^{(1)}$, $P_4 = x^{(2)}$, $P_2 = x^{(3)}$, $P_3 = x^{(4)}$ the current population I_0 (Fig. 16.6) and solve the problem

$$x^B = \arg\max_{t\in[0,1]} f\left[\frac{1}{2}\left[-t(1-t)^2 P_1 + \left(2 - 5t^2 + 3t^3\right) P_2 + t\left(1 + 4t - 3t^2\right) P_3 - t^2(1-t) P_4\right]\right]. \tag{16.17}$$

Step 6.5 Reduce the population. Place individuals of the current population, replenished with $x^{\text{Bezier3}}, x^{CR}, x^{\text{Bezier4}}, x^B$ members increasing the value of the fitness function. Leave only NP best.

Step 7 Reduce the population.

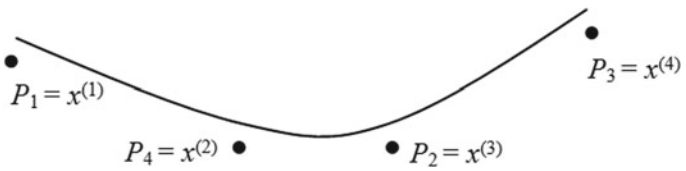


Fig. 16.6 Three point of B-spline

Place individuals of the current population increasing the value of the fitness function $I_t = \{x^{(1)}, \ldots, x^{(NP)}\}$, where $NP = b_1 + b_2$, $f(x^{(1)}) = f_{\max}$. Delete the last b_2 individuals (with the worst value of the fitness function).

Increase the number of iterations $it = it + 1$.

The result of Step 7 is a reduced population.

Step 8 Execute the replenishment of population.

Step 8.1 Perform this step b_2 times. Generate a population consisting of b_2 individuals on a set D using a uniform distribution: $x^1, \ldots, x^{b_2}$.

Step 8.2 Order the individuals of the population in ascending order of the fitness function value: $I_t = \{x^{(1)}, \ldots, x^{(NP)}\}$, where $NP = b_1 + b_2$, $f(x^{(1)}) = f_{\max}$.

Step 9 Check the stop conditions of global search.

If $it < I_{\max}$, then continue search, go to Step 4.

If $it \geq I_{\max}$, then finish search, go to Step 10.

Step 10 Select the solution from the last population.

Stop the algorithm. As an approximate solution to the problem $f(x^*) = \max\limits_{x \in D} f(x)$, select individual with the greatest value of the fitness function from the current population: $x^* \cong \tilde{x}^* = \arg\max\limits_{j=1,\ldots,NP} f(x^j)$.

Recommendations on the parameters selection. Size of the population NP determines a number of calculations of the objective function at each iteration. For a task with a large range of feasible solutions, it is recommended to take a larger parameter value *NP*. Recommended value for this parameter is $NP \in [30, 40]$.

The number of iterations $I_{\max}$ determines how long the search for new solutions will continue. The more $I_{\max}$, the more accurate the solution could be found. Recommended values for the considered set of standard functions depending on the complexity of the function are $I_{\max} \in [100,\ 300]$.

The number of points M_1 obtained using the Bezier curves in the interpolation search phase. When maximizing the objective function along the parametric curve, M_1 agents are formed. Recommended value for this parameter is $M_1 \in [2,\ 5]$.

The number of points M_2 obtained using B-spline during the interpolation search phase. When maximizing the objective function along the parametric curve, M_2 agents are formed. This curve is used to explore new areas. Recommended value for this parameter is $M_2 \in [4,\ 6]$.

Parameter PRT determines the activity of searching by coordinate during migration of group agents. The parameter sets the search direction. Recommended value for this parameter is $PRT \in [0.005, 0.06]$.

The number $n$step defines the possible positions of the population members during the migration of the population. The parameter determines how many steps an individual will take in the direction to the leader. Recommended value for this parameter is $n\text{step} \in [3, 7]$.

Number b_2 helps to reduce the population individuals (worst). The worst b_2 individuals are removed from the population, and new b_2 individuals are added to maintain the same number of individuals NP. Recommended value for this parameter is $b_2 \in [4,\ 8]$.

16.2.3 Multi-agent Optimization Algorithm Using Linear Regulators for Agents Motion Control

Solution search strategy. It is required to generate a population of NP agents on a set of admissible solutions D using a uniform distribution law. The search for an extremum is realized in a given number of passes $P_{\max}$. In the next pass, all agents move under the action of appropriate control for a certain number of iterations.

Let suppose the equation of the agent motion of the form of Eq. 16.18, where x is n-dimensional vector of agent position, v is n-dimensional vector of agent velocity, t is the time, t_0 is the initial time in the next passage, x_0 is the initial position, v_0 is the initial velocity, u is n-dimensional vector of agent control.

$$\begin{aligned} \frac{\mathrm{d}x}{\mathrm{d}t} &= v \quad x(t_0) = x_0 \\ \frac{\mathrm{d}v}{\mathrm{d}t} &= u \quad v(t_0) = v_0 \end{aligned} \tag{16.18}$$

As $t_0 = 0$ let $v_0 = o$ (o is zero n-dimensional vector).

Denote $X = \begin{pmatrix} x \\ v \end{pmatrix}$ as the extended state vector of the agent and rewrite Eq. 16.2 in the form

$$\frac{\mathrm{d}\begin{pmatrix} x \\ v \end{pmatrix}}{\mathrm{d}t} = \begin{pmatrix} O_n & E_n \\ O_n & O_n \end{pmatrix}\begin{pmatrix} x \\ v \end{pmatrix} + \begin{pmatrix} O_n \\ E_n \end{pmatrix} u \tag{16.19}$$

or

$$\frac{\mathrm{d}X}{\mathrm{d}t} = AX(t) + B\,u(t), \quad X(t_0) = X_0 = \begin{pmatrix} x_0 \\ v_0 \end{pmatrix}, \tag{16.20}$$

where O_n, E_n are the zero and unit matrices of order n, $A = \begin{pmatrix} O_n & E_n \\ O_n & O_n \end{pmatrix}$, $B = \begin{pmatrix} O_n \\ E_n \end{pmatrix}$ are the matrices of size $(2n \times 2n), (2n \times n)$, respectively. No restrictions are imposed on the control vector, i.e., $u \in R^n$.

In the initial population ($k = 0$), as well as, at the end of each kth pass, determine the position of the leader among the agents of the population and the corresponding

maximum value of the objective function $x^{\text{best},k}$, $f(x^{\text{best},k})$. During the next pass, it does not change the position according to Eqs. 16.21, where the control of leader $u^{\text{best}} = o$ or $\frac{\mathrm{d}X^{\text{best}}}{\mathrm{d}t} = o$ with leader state vector $X^{\text{best}} = (x^{\text{best}}, v^{\text{best}})^T$.

$$\begin{aligned} &\frac{\mathrm{d}x^{\text{best}}}{\mathrm{d}t} = o \quad x^{\text{best}}(t_k) = x^{\text{best},k} \\ &\frac{\mathrm{d}v^{\text{best}}}{\mathrm{d}t} = o \quad v^{\text{best}}(t_k) = o \\ &\qquad t \in [t_k, t_{k+1}) \quad k = 0, \ldots, P_{\max} - 1 \end{aligned} \tag{16.21}$$

The remaining $(NP - 1)$ agents are divided into four equal groups:

- Agents using the minimization criterion of agents' movement to the current leader in a finite time interval.
- Agents using the minimization criterion of agents' movement to the current leader in an infinite time interval.
- Agents using the minimization semi-defined criterion (so-called criterion of the generalized work)—control agents for a finite time interval.
- Agents using functional increment minimization of agents' movement to the current leader at the current moment (locally optimal approach).

Dividing into four groups is optional, so all agents can be placed in one group or divided into two or three groups, and during the calculation, a set of agents can be divided in different ways. In this version of the algorithm, the division was carried out into four groups.

For all agents of each group, the positions and velocity vectors are different, but the same feedback control law could be found and applied determined by the relation for a linear optimal controller, the gain matrix of which is found from the minimization condition of the quadratic control quality criterion characterizing the nature of the approximation agent to agent leader at the current iteration, as well as, the intensity of the control signal applied.

Introduce the deviation from the leader $\Delta X = X - X^{\text{best}}$, whose change is described by Eq. 16.22 (subtracting Eq. 17.21 from Eq. 17.20), where x^k, v^k are the agent position and velocity at the end of the previous pass, respectively.

$$\begin{aligned} &\frac{\mathrm{d}\Delta X}{\mathrm{d}t} = A\,\Delta X(t) + B\,u(t) \quad \Delta X(t_k) = \begin{pmatrix} x^k - x^{\text{best},k} \\ v^k \end{pmatrix} \\ &\qquad t \in [t_k, t_{k+1}] \quad k = 0, \ldots, P_{\max} - 1 \end{aligned} \tag{16.22}$$

Movement of the first group of agents (for all agents of the group, optimal control with a finite horizon is applied) is simulated as follows. Quality criterion for controlling the agents' trajectories of the first group has a view of Eq. 16.23, where Λ, $S(t)$ is the non-negative definite symmetric matrices of size $(n \times n)$, $Q(t)$ is the positive definite symmetric matrix $(n \times n)$.

$$I = \frac{1}{2}\int_{t_k}^{t_{k+1}} \left[\Delta X^T(t) S(t) \Delta X(t) + u^T(t) Q(t) u(t)\right] \mathrm{d}t + \frac{1}{2}\Delta X^T(t_{k+1}) \Lambda \Delta X(t_{k+1}) \tag{16.23}$$

For any initial states, optimal feedback control $u^*(t, \Delta X)$ has the form:

$$u^*(t, \Delta X) = -Q^{-1}(t) B^T(t) P(t) \Delta X = -F(t) \Delta X, \tag{16.24}$$

where matrix coefficients of the gain of the linear optimal regulator $F(t) = Q^{-1}(t) B^T(t) P(t)$, $P(t)$ is the symmetric matrix of sizes $(n \times n)$ satisfying the Riccati differential equation provided by Eq. 16.25.

$$\dot{P}(t) = -A^T(t) P(t) - P(t) A(t) + P(t) B(t) Q^{-1}(t)\, B^T(t) P(t) - S(t), \quad P(t_{k+1}) = \Lambda \tag{16.25}$$

Here, $t \in [t_k, t_{k+1}]$, $t_0 = 0$, value $t_{k+1} = t_k + N\,\mathrm{MAX} \cdot h$, where $N\,\mathrm{MAX}$ is the given number of iterations, h is the integration step. To simplify the solution, one can assume everywhere $A(t) = A$, $B(t) = B$, $Q(t) = Q$, $S(t) = S$ because the models provided by Eqs. 16.20–16.21 are linear stationary.

Movement of the second group of agents (for all agents of the group, optimal control with an infinite horizon is applied) is simulated as follows. The quality criterion for controlling the trajectories of agents of the second group has a view of Eq. 16.26, where S is the non-negative definite symmetric numerical matrix of sizes $(n \times n)$, Q is the positive definite symmetric numerical matrix $(n \times n)$.

$$I = \frac{1}{2}\int_{t_k}^{+\infty} \left[\Delta X^T(t) S \Delta X(t) + u^T(t) Q u(t)\right] \mathrm{d}t \tag{16.26}$$

For any initial states, optimal feedback control $u^*(\Delta X)$ has the form

$$u^*(\Delta X) = -Q^{-1} B^T P \Delta X = -F \Delta X, \tag{16.27}$$

where matrix coefficients of the gain of the linear optimal regulator $F = Q^{-1} B^T P$, P is the positive definite symmetric matrix satisfying the Riccati algebraic equation provided by Eq. 16.28.

$$-A^T P - P A + P B Q^{-1} B^T P - S = 0 \tag{16.28}$$

The solution to this equation satisfying the Sylvester criterion is unique.

Movement of the third group of agents (for all agents of the group, optimal control is applied according to the criterion of generalized work) is simulated as follows. The quality criterion for controlling the agents' trajectories of the third group is provided

by Eq. 16.29, where Λ, $S(t)$ is the non-negative definite symmetric matrices of sizes $(n \times n)$, $Q(t)$ is the positive definite symmetric matrix $(n \times n)$.

$$I^{\text{o.p.}} = \frac{1}{2}\int_{t_k}^{t_{k+1}} \left[\Delta X^T(t)S(t)\Delta X(t) + u^T(t)Q(t)u(t) + \Delta X^T(t)P(t)B(t)Q^{-1}(t)B^T(t)P(t)\Delta X(t)\right]dt + \frac{1}{2}\Delta X^T(t_{k+1})\Lambda\Delta X(t_{k+1}) \tag{16.29}$$

For any initial states, optimal feedback control with $u^{\text{o.p.}}(t, \Delta X)$ has the form:

$$u^{o.p.}(t, \Delta X) = -Q^{-1}(t)B^T(t)P(t)\Delta X = -F(t)\Delta X, \tag{16.30}$$

where matrix coefficients of the gain of the linear regulator $F(t) = Q^{-1}(t)B^T(t)P(t)$, $P(t)$ is the symmetric matrix satisfying the linear differential equation provided by Eq. 16.31.

$$\dot{P}(t) = -A^T(t)\,P(t) - P(t)\,A(t) - S(t) \quad P(t_{k+1}) = \Lambda \tag{16.31}$$

Movement of the fourth group of agents (for all group agents, locally optimal control is applied) is simulated as follows. The quality criterion for controlling the trajectories of agents of the fourth group has the form of Eq. 16.32, where $\Lambda(t)$, $S(t)$ is the non-negative definite symmetric matrices of sizes $(n \times n)$, $Q(t)$ is the positive definite symmetric matrix $(n \times n)$.

$$I^{\text{loc}} = \frac{1}{2}\int_{t_k}^{t} \left[\,\Delta X^T(\tau)S(\tau)\Delta X(\tau) + u^T(\tau)Q(\tau)u(\tau)\right]\mathrm{d}\tau + \frac{1}{2}\Delta X^T(t)\Lambda(t)\Delta X(t) \tag{16.32}$$

For any initial state, the locally optimal feedback control $u^{\text{loc}}(t, \Delta X)$ has the form of Eq. 16.33, where matrix coefficients of the gain of the linear regulator $F(t) = Q^{-1}(t)B^T(t)\Lambda(t)$.

$$u^{\text{loc}}(t, \Delta X) = -Q^{-1}(t)B^T(t)\Lambda(t)\Delta X = -F(t)\Delta X \tag{16.33}$$

After performing NMAX iterations, each agent presents the best result obtained during the movement. This completes the next passage.

Then, among all the agents of the population, the leader is again selected and a new division into groups is made. The passage repeat process ends when the maximum number of passes is reached.

Comment. For simplicity suppose that

$$S(t) = \begin{pmatrix} k_S E_n & O_n \\ O_n & O_n \end{pmatrix}, \quad Q(t) = E_n,$$

$$\Lambda = \begin{pmatrix} k_\Lambda E_n & O_n \\ O_n & O_n \end{pmatrix}, \quad k_S > 0, \quad k_\Lambda > 0.$$

Solution search algorithm

Step 1 Set the method parameters: NP is the number of agents in the population, $P_{\max}$ is the maximum number of passes, $NMAX$ is the number of iterations per pass, h is the step of integrating differential equations, and k_S, k_Λ are the coefficients for matrices defining quality criteria. Let $k = 0$ (iteration count), $v^0 = o$, $t_0 = 0$.

Step 2 Generate the initial population on set D using the uniform distribution law: $x^1, \ldots, x^{NP}$. Calculate the values of the objective function $f(x^1), \ldots, f(x^{NP})$.

Step 3 Order the population consisting of NP agents by the value of the objective function.

Step 4 Choose a leader agent and the corresponding best objective function value:$x^{\text{best},k}$, $f^{\text{best},k}$.

Step 5 Divide all other $(NP - 1)$ agents arbitrarily into four groups.
For each agent, create a differential equation

$$\frac{\mathrm{d}\Delta X}{\mathrm{d}t} = A\Delta X(t) + B\,u(t), \quad \Delta X(t_k) = \begin{pmatrix} x^k - x^{\text{best},k} \\ v^k \end{pmatrix} \tag{16.34}$$

where

$$A = \begin{pmatrix} O_n & E_n \\ O_n & O_n \end{pmatrix}, B = \begin{pmatrix} O_n \\ E_n \end{pmatrix}$$

Step 6 Move the agents of the first group (optimal control with finite horizon is applied).

Step 6.1 Find the solution of the Riccati differential equation $P(t), \forall t \in [t_k, t_{k+1}]$ using Eq. 16.35, where $t_{k+1} = t_k + NMAX \cdot h$.

$$\dot{P}(t) = -A^T(t)P(t) - P(t)A(t) + P(t)B(t)Q^{-1}(t)B^T(t)P(t) - S(t), \quad P(t_{k+1}) = \Lambda \tag{16.35}$$

Step 6.2 Find the optimal feedback control $u^*(t, \Delta X)$ using Eq. 16.36, where $F(t) = Q^{-1}(t)B^T(t)P(t)$.

$$u^*(t, \Delta X) = -Q^{-1}(t)B^T(t)P(t)\Delta X = -F(t)\Delta X \tag{16.36}$$

Step 6.3 For each agent of the first group, execute:

Step 6.3.1 Find the solution of differential equation in the form of Eq. 16.37 on time interval $[t_k, t_{k+1}]$: $\Delta X(t_i), t_i = t_k + ih, \quad i = 0, 1, \ldots, NMAX - 1.$

$$\frac{\mathrm{d}\Delta X}{\mathrm{d}t} = A\Delta X(t) + B\,u^*(t, \Delta X(t)), \quad \Delta X(t_k) = \begin{pmatrix} x^k - x^{\mathrm{best},k} \\ v^k \end{pmatrix} \tag{16.37}$$

Step 6.3.2 Find the agent state vectors during a pass using Eq. 16.38, where $X^{I,\mathrm{New}} \in [a_i, b_i],\ i = 1, ..., n.$

$$X^{\mathrm{I,New}}(t_i) = X^{\mathrm{best}} + \Delta X(t_i), \quad i = 0, 1, \ldots, NMAX - 1 \tag{16.38}$$

Step 6.3.3 Among all the positions of the agent during the passage, choose the best for the entire period of movement, which corresponds to the best value of the objective function.

Step 7 Move the agents of the second group (optimal control with an infinite horizon is applied).

Step 7.1 Find the solution of the Riccati algebraic equation provided by Eq. 16.39.

$$-A^T P - PA + PBQ^{-1}B^T P - S = 0 \tag{16.39}$$

Step 7.2 Find the optimal feedback control $u^*(\Delta X)$ using Eq. 16.40 where $F = Q^{-1}B^T P$.

$$u^*(\Delta X) = -\ Q^{-1}B^T P\ \Delta X = -F\ \Delta X \tag{16.40}$$

Step 7.3 For each agent of the second group, execute:

Step 7.3.1 Find the solution of differential equation provided by Eq. 16.41.

$$\frac{\mathrm{d}\Delta X}{\mathrm{d}t} = A\Delta X(t) + Bu^*(\Delta X(t)), \quad \Delta X(t_k) = \begin{pmatrix} x^k - x^{\mathrm{best},k} \\ v^k \end{pmatrix} \tag{16.41}$$

Step 7.3.2 Find the agent state vectors during a pass by Eq. 16.42, where $X^{\mathrm{II,New}} \in [a_i, b_i], \quad i = 1, \ldots, n.$

$$X^{\mathrm{II,New}}(t_i) = X^{\mathrm{best}} + \Delta X(t_i) \quad i = 0, 1, \ldots, N\mathrm{MAX} - 1 \tag{16.42}$$

Step 7.3.3 Among all the positions of the agent during the passage, choose the best for the entire period of movement, which corresponds to the best value of the objective function.

Step 8 Move the agents of the third group (optimal control according to the criterion of generalized work is applied).

Step 8.1 Find the solution of linear differential equation $P(t)$, $\forall t \in [t_k, t_{k+1}]$ using Eq. 16.43.

$$\dot{P}(t) = -A^T(t)P(t) - P(t)A(t) - S(t), \quad P(t_{k+1}) = \Lambda \tag{16.43}$$

Step 8.2 Find the optimal feedback control $u^{\text{o.p.}}(t, \Delta X)$ using Eq. 16.44 where $F(t) = Q^{-1}(t)B^T(t)P(t)$.

$$u^{\text{o.p.}}(t, \Delta X) = -Q^{-1}(t)B^T(t)P(t)\Delta X = -F(t)\Delta X \tag{16.44}$$

Step 8.3 For each agent of the third group, execute:

Step 8.3.1 Find the solution of differential equation using Eq. 16.45.

$$\frac{\mathrm{d}\Delta X}{\mathrm{d}t} = A\Delta X(t) + Bu^{\text{o.p.}}(t, \Delta X(t)), \quad \Delta X(t_k) = \begin{pmatrix} x^k - x^{\text{best},k} \\ v^k \end{pmatrix} \tag{16.45}$$

Step 8.3.2 Find the agent state vectors during a pass using Eq. 16.46 where $X^{\text{III,New}} \in [a_i, b_i]$, $i = 1, \ldots, n$.

$$X^{\text{III,New}}(t_i) = X^{\text{best}} + \Delta X(t_i) \quad i = 0, 1, \ldots, N\text{MAX} - 1 \tag{16.46}$$

Step 8.3.3 Among all the positions of the agent during the passage, choose the best for the entire period of movement, which corresponds to the best value of the objective function.

Step 9 Move the agents of the fourth group (locally optimal control is applied).

Step 9.1 Find the locally optimal feedback control $u^{\text{loc}}(t, \Delta X)$ using Eq. 16.47, where $F(t) = Q^{-1}(t)B^T(t)\Lambda(t)$.

$$u^{\text{loc}}(t, \Delta X) = -Q^{-1}(t)B^T(t)\Lambda(t)\Delta X = -F(t)\Delta X \tag{16.47}$$

Step 9.2 For each agent of the fourth group, execute:

Step 9.2.1 Find the solution of differential equation using Eq. 16.48.

$$\frac{\mathrm{d}\Delta X}{\mathrm{d}t} = A\Delta X(t) + Bu^{\text{loc}}(t, \Delta X(t)), \quad \Delta X(t_k) = \begin{pmatrix} x^k - x^{\text{best},k} \\ v^k \end{pmatrix} \tag{16.48}$$

Step 9.2.2 Find the agent state vectors during a pass using Eq. 16.49 where $X^{\text{IV,New}} \in [a_i, b_i]$, $i = 1, \ldots, n$.

$$X^{\text{IV,New}}(t_i) = X^{\text{best}} + \Delta X(t_i) \quad i = 0, 1, \ldots, N\text{MAX} - 1 \tag{16.49}$$

Step 9.2.3 Among all the positions of the agent during the passage, choose the best for the entire period of movement, which corresponds to the best value of the objective function.
The result of Steps 6–8 are the positions of NP agents in the population, of which $(NP-1)$ agents with a new position as a result of movement under the action of control within a fixed group and the leader agent x^{best}, which has not changed its position during the last pass.

Step 10 Check the global search stop conditions.
If $k < P_{\max} - 1$, then continue search, go to Step 3 (order the population of agents, identify the current leader, shuffle groups) and let $k = k + 1$.
If $k \geq P_{\max} - 1$, then finish search, go to Step 11.

Step 11 Select the solution from the last population.
Stop the algorithm. As an approximate solution to the problem $f(x^*) = \max\limits_{x \in D} f(x)$ select the agent with the best value of the objective function from the current population $x^* \cong \tilde{x}^* = \arg\max\limits_{j=1,\ldots,NP} f(x^j)$.

Recommendations on the parameters selection. The size of population NP determines the number of calculations of the objective function at each iteration. The value $NP - 1$ must be divisible by four. For a task with a large range of admissible solutions, it is recommended to take a larger value of the parameter NP. Recommended parameter value is $NP \in [501, 1601]$.

The number of iterations $NMAX$ during the passage determines how long the search for new solutions per passage will continue. As the parameter NMAX increases, the accuracy of the solution increases. Recommended values for the considered set of standard functions depending on the complexity of the function are $NMAX \in [20, 80]$.

The maximum number of passes $P_{\max}$ determines how long the search for new solutions will continue. As the number of passes increases, the accuracy of the solution increases. Recommended value for this parameter is $P_{\max} \in [10,\ 30]$.

Coefficient k_S for matrix S defines the criterion of quality of agent trajectories control. Recommended value for this parameter is $k_S \in [0.1,\ 1]$.

Coefficient k_Λ for matrix Λ defines the criterion of quality of agent trajectories control. Recommended value for this parameter is $k_\Lambda \in [1,\ 5]$.

Step of integrating Riccati differential equations is denoted by h. Recommended value for this parameter is $h = 0.0001$.

16.3 Application of Multi-agent Methods for Optimal Open-Loop Control Problems

Hereinafter, Sect. 16.3.1 involves a statement of the problem. Search algorithm of optimal open-loop control using switching points is presented in Sect. 16.3.2.

Section 16.3.3 develops search algorithm of optimal open-loop control using expansion in a system of basis functions. Solving the problem of finding optimal open-loop control is considered in Sect. 16.3.4.

16.3.1 Statement of the Problem

Let the behavior of the control object model be described by an ordinary differential equation in the form of Eq. 16.50, where x is the system state vector, $x = (x_1, \ldots, x_n)^T \in R^n$, u is the control vector, $u = (u_1, \ldots, u_q)^T \in U \subseteq R^q$, U is some given set of admissible control values determined by the direct product of segments $[a_1, b_1] \times \cdots \times [a_q, b_q]$, $t \in T = [t_0,\ t_1]$ is the time interval, the start time t_0 and terminal time t_1 are given, $f(t, x, u)$ is the continuous vector function; R^n is n-dimensional Euclidean space.

$$\dot{x}(t) = f(t, x(t), u(t)) \tag{16.50}$$

The initial condition $x(t_0) = x_0$ sets the initial state of the system.

We define the set of admissible processes $D(t_0, x_0)$ as a set of pairs $d = (x(\cdot), u(\cdot))$ that include the trajectory $x(\cdot)$ and control $u(\cdot)$ (where $\forall t \in T : x(t) \in R^n,\ u(t) \in U$, functions $x(\cdot)$ are continuous and piecewise-differentiable, and $u(\cdot)$ piecewise-continuous) satisfying Eq. 16.50 with given initial condition.

On the set $D(t_0, x_0)$, we define the cost functional in the form of Eq. 16.51.

$$I(d) = F(x(t_1)) \tag{16.51}$$

It is need to find such a pair $d^* = (x^*(\cdot), u^*(\cdot)) \in D(t_0, x_0)$ that $I(d^*) = \min\limits_{d \in D(t_0, x_0)} I(d)$.

16.3.2 Search Algorithm of Optimal Open-Loop Control Using Switching Points

We consider Eq. 16.50 as a linear in control, which has the form of Eq. 16.52, where $A(x)$ is the nonlinear function and $B(t)$ is the matrix $(n \times q)$ depending on time.

$$\dot{x}(t) = A(x(t)) + B(t)u(t) \tag{16.52}$$

In Eq. 16.52, the structure of optimal open-loop control is relay according to the maximum principle; therefore, it is proposed to look for an approximate solution in a parametric form determined by the number of control switching moments and their values.

The search algorithm of optimal open-loop control using switching points is the following.

Step 1 Initialization. Select a method from the group of multi-agent algorithms and set its parameters. Set the number of switching $p = 0$ in the control $u(t)$; wherein $t_{\Pi_0} \in \{t_0, t_1\}$.

Step 2 Generate the initial population (controls) of NP individuals on the time interval $t \in [t_0, t_1]$. The resulting sequences of values $1, \ldots, NP$ are the switching points $t_\Pi \in [t_0, t_1]$ in the control $u(t)$.

Step 3 Generate control by generating the switching point values

$$u_p^j(t) = a_p \chi(t_0) + (a_p - b_p) \sum_{k=0}^{p} (-1)^k \chi(t - t_{\Pi_k}), \tag{16.53}$$

where

$$\chi(t) = \begin{cases} 0 \; t \le 0 \\ 1 \; t > 0 \end{cases}, \quad j \in \overline{1, NP}, \quad p \in \overline{1, q}, \quad a_p \le u \le b_p.$$

Step 4 Integrate NP systems of differential equations (Eq. 16.52) with controls $u^1(t), \ldots, u^{NP}(t)$ using the fourth-order Runge–Kutta method. For any individual, obtain the corresponding trajectories $x_1^1, \ldots, x_1^{NP}, \ldots, x_n^1, \ldots, x_n^{NP}$ and calculate the values of the cost functional $I^1, \ldots, I^{NP}$.

Step 5 Fulfill the next iteration of the selected method of minimizing the functional (Eq. 16.51). Obtain new positions of individuals $1', \ldots, NP'$ (switching point values). Go to Step 3.

Step 6 The loop (Step 3–Step 5) ends when a certain number of iterations are reached. The best individual is selected (set of control switching points). The corresponding control and trajectory, as well as, the value of the cost functional I_p^*, are taken as an approximate solution of the problem with the number of switching equal to p.

Step 7 If $I_p^* < I_{p-1}^*$ (condition is checked under $p \ge 1$), then let $p = p + 1$ and go to Step 2. If $I_p^* \ge I_{p-1}^*$, then the search procedure for optimal open-loop control is completed and control with p switching is selected.

16.3.3 Search Algorithm of Optimal Open-Loop Control Using Expansion in a System of Basis Functions

A class of nonlinear continuous deterministic dynamical systems linear in bounded control is considered in the form of Eq. 16.54.

$$\dot{x}(t) = A(x) + B(x)u \tag{16.54}$$

The quality criterion is set by the Mayer functional (Eq. 16.51). The desired optimal open-loop control is sought in the form of a saturation function, which should guarantee the fulfillment of the parallelepiped type constraints on control vector. The saturation function has a relay structure, and it is proposed to search for its arguments in the form of a linear combination of given basis functions [12, 13].

The search algorithm of optimal open-loop control using expansion in a system of basis functions is the following.

Step 1 Initialization. Select a method from the group of multi-agent algorithms and set its parameters. Set the initial time truncation scale $L = 1$, the range of possible values of the decomposition coefficient $c_0 \in \left[c_{0_1}, c_{0_2}\right]$.

Step 2 Generate the initial population (controls) of NP individuals, which determinate by coefficients c_i of expansion $g(t)$, using Eq. 16.55 where $c_i \in \left[c_{i_1}, c_{i_2}\right]$, $i \in \overline{0, L-1}$.

$$\left\{c_0^{(j)}, c_1^{(j)}, \ldots, c_{L-1}^{(j)}\right\}, \quad j \in \overline{1, NP} \tag{16.55}$$

Step 3 Using the generated coefficients, form the control in the form of a saturation function sat that guarantees the fulfillment the constraints on control vector:

$$u_j^{(m)}(t) = \text{sat}\left\{g_j(t)\right\}, \quad j \in \overline{1, q}, \tag{16.56}$$

where

$$\forall t \in T, \quad \text{sat}\, g_j(t) = \begin{cases} a_j & g_j(t) \le 0 \\ b_j & g_j(t) > 0 \end{cases}, \quad g_j(t) = \sum_{i=0}^{L-1} c_i^{(j)} p_i(t).$$

As the basis function $p_i(t)$, we can take the system of nonstationary cosine curves orthonormalized on the time interval $T = [t_0, t_1]$ with $t_0 = 0$ in the form of Eq. 16.57.

$$p_i(t) = \begin{cases} \sqrt{\frac{1}{t_1}} & i = 0 \\ \sqrt{\frac{2}{t_1}} \cos\left(\frac{i\pi t}{t_1}\right) & i = 1, 2, \ldots, L-1 \end{cases} \tag{16.57}$$

Step 4 Integrate NP systems of differential equations (Eq. 16.52) with controls $u^1(t), \ldots, u^{NP}(t)$ using the 4th order Runge–Kutta method. For any individual, obtain the corresponding trajectories $x_1^1, \ldots, x_1^{NP}, \ldots, x_n^1, \ldots, x_n^{NP}$ and calculate the values of the cost functional $I^1, \ldots, I^{NP}$.

Step 5 Fulfill the next iteration of the selected method of minimizing the functional (Eq. 16.51). Obtain new positions of individuals $1', \ldots, NP'$(coefficient values). Go to Step 3.

Table 16.1 Formulation of task 1

The dimension of the state vector	$n = 2$
Time interval	$t \in [0, 1]$
Control constraint	$-1 \le u \le 1$
Initial value	$x(0) = (0, 0)$
System of differential equations	$\begin{cases} \dot{x}_1 = x_2 + \sin x_1 + u \\ \dot{x}_2 = x_1 \cos x_2 u \end{cases}$
Cost functional	$I(u) = x_2(1)$

Step 6 The loop (Step 3–Step 5) ends when a certain number of iterations are reached. The best individual is selected (set of coefficients c_i). The corresponding control and trajectory, as well as, the value of the functional $I^*_{c_i}$ are taken as an approximate solution to the problem with the found coefficients c_i^{*L}, $i \in \overline{0, L-1}$ with a given truncation scale L.

Step 7 If $I^*_{c_i^L} < I^*_{c_i^{L-1}}$ (condition is checked under $L \ge 1$), then let $L = L + 1$ and go to Step 2. If $I^*_{c_i^L} \ge I^*_{c_i^{L-1}}$, then the search procedure for optimal open-loop control is completed and control with c_i^{*L-1} coefficients is selected.

16.3.4 Solving the Problem of Finding Optimal Open-Loop Control

Task 1. Formulation of the task (Table 16.1) [14, 15].

Solving Task 1 by the search algorithm of optimal open-loop control using switching points. The best number of switches: $p = 1$.

Optimization method and its parameters: hybrid multi-agent optimization method of interpolation search ($NP = 30$, $I_{\max} = 50$, $M_1 = 2$, $M_2 = 5$, $PRT = 0.01$, $n$step $= 5$, an $b_2 = 8$) and multi-agent optimization algorithm using linear regulators for agents motion control ($NP = 101$, $NMAX = 50$, $P_{\max} = 10$, $k_S = 0.1$, $k_\Lambda = 5$, and $h = 0.0001$).

The results of solving Task 1 with the help of search algorithm of optimal open-loop control using switching points are presented in Table 16.2.

Solving Task 1 by the search algorithm of optimal open-loop control using expansion in a system of basis functions. The best number of coefficients in expansion: $L = 2$.

Optimization method and its parameters: hybrid multi-agent optimization method of interpolation search ($NP = 30$, $I_{\max} = 50$, $M_1 = 2$, $M_2 = 5$, $PRT = 0.01$, $n$step $= 5$, and $b_2 = 8$) and multi-agent optimization algorithm using linear regulators for agents motion control ($NP = 41$, $NMAX = 40$, $P_{\max} = 20$, $k_S = 1$, $k_\Lambda = 5$, and $h = 0.0001$).

The results of solving Task 1 by the search algorithm of optimal open-loop control using expansion in a system of basis functions are presented in Table 16.3.

Table 16.2 Results of solving task 1

Optimization method	Coordinates of points $(x_1(1),\ x_2(1))$	Switching coordinate	The value of the functional I
Hybrid multi-agent optimization method of interpolation search	(0.444665, −0.13598)	0.5	−0.13598
Multi-agent optimization algorithm using linear regulators for agents motion control	(0.44999, −0.13450)	0.5	−0.134550
Known solution [15]	(0.440804, −0.13593)	0.5	−0.13599

Table 16.3 Results of solving task 1

Optimization method	Coordinates of points $(x_1(1), x_2(1))$	Coefficients in expansion c_i	The value of the functional I
Hybrid multi-agent optimization method of interpolation search	(0.55055, −0.13349)	5.05, 6.51	−0.13598
Multi-agent optimization algorithm using linear regulators for agents motion control	(0.57325, −0.13208)	4.63, 5.66	−0.13208

Graphs of optimal trajectories and control are shown in Fig. 16.7.

Task 2. Formulation of the task (Table 16.4) [14, 15].

Solving Task 2 by the search algorithm of optimal open-loop control using switching points. The best number of switches: $p = 4$.

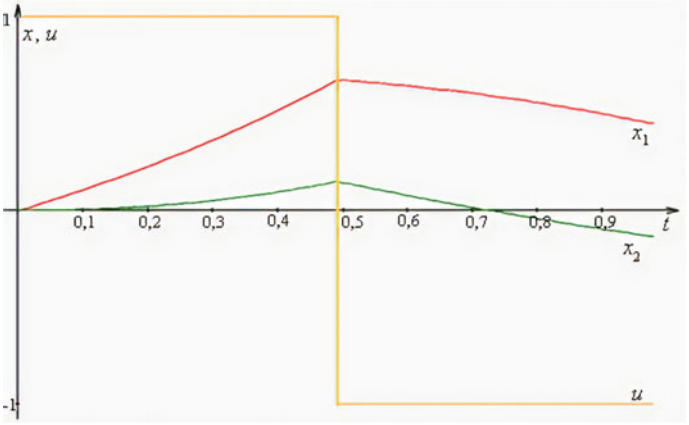


Fig. 16.7 Trajectories x_1 and x_2 and control u for task 1

Table 16.4 Formulation of task 2

The dimension of the state vector	$n = 2$
Time interval	$t \in [0, 2]$
Control constraint	$-1 \le u \le 2$
Initial value	$x(0) = (-1, 0)^T$
System of differential equations	$\begin{cases} \dot{x}_1 = x_2^2 + u \\ \dot{x}_2 = 8 \sin x_1 + x_1 - x_2 - u \end{cases}$
Cost functional	$I(u) = -x_2(2)$

Table 16.5 Results of solving task 2

Optimization method	Coordinates of points $(x_1(2),\ x_2(2))$	Switching point coordinate	The value of the functional I
Hybrid multi-agent optimization method of interpolation search	(16.36429, 6.06547)	(0.58, 1.35, 1.54, 1.77)	−16.36429
Multi-agent optimization algorithm using linear regulators for agents motion control	(16.66516, 6.52485)	(0.59, 1.29, 1.53, 1.97)	−16.66516
Known solution [15]	(16.76268, 6.35095)	(0.5, 1.25, 1.5, 1.8)	−16.76268

Optimization method and its parameters: hybrid multi-agent optimization method of interpolation search ($NP = 30$, $I_{\max} = 200$, $M_1 = 2$, $M_2 = 5$, $PRT = 0.01$, $n\text{step} = 5$, and $b_2 = 8$) and multi-agent optimization algorithm using linear regulators for agents motion control ($NP = 801$, $NMAX = 50$, $P_{\max} = 10$, $k_S = 0.1$, $k_\Lambda = 5$, and $h = 0.0001$).

The results of solving Task 2 by the search algorithm of optimal open-loop control using switching points are presented in Table 16.5.

Solving Task 2 by the search algorithm of optimal open-loop control using expansion in a system of basis functions. The best number of coefficients in expansion: $L = 4$.

Optimization method and its parameters: hybrid multi-agent optimization method of interpolation search ($NP = 40$, $I_{\max} = 400$, $M_1 = 2$, $M_2 = 5$, $PRT = 0.01$, $n\text{step} = 5$, and $b_2 = 8$) and multi-agent optimization algorithm using linear regulators for agents motion control ($NP = 401$, $NMAX = 40$, $P_{\max} = 10$, $k_S = 1$, $k_\Lambda = 5$, and $h = 0.0001$).

The results of solving Task 2 by the search algorithm of optimal open-loop control using expansion in a system of basis functions are presented in Table 16.6.

Graphs of optimal trajectories and controls are shown in Fig. 16.8.

Task 3. Formulation of the task (Table 16.7) [14, 15].

Solving Task 3 by the search algorithm of optimal open-loop control using switching points. The best number of switches: $p = 1$.

Table 16.6 Results of solving task 2

Optimization method	Coordinates of points $(x_1(2),\ x_2(2))$	Coefficients in expansion c_i	The value of the functional I
Hybrid multi-agent optimization method of interpolation search	(13.01829, 4.44509)	−0.23, 1.73, 1.78, 1.81	−13,01,829
Multi-agent optimization algorithm using linear regulators for agents motion control	(12.36497, 4.16002)	−0.28, 1.61, 1.79, 1.92	−12,36,497

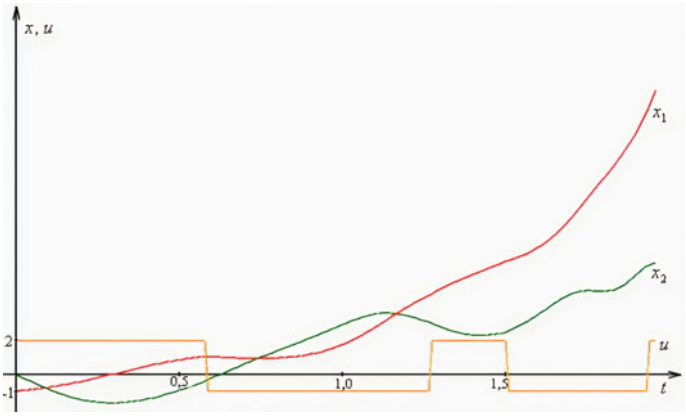


Fig. 16.8 Trajectories x_1 and x_2 and control u for task 2

Table 16.7 Formulation of task 3

The dimension of the state vector	$n = 2$
Time interval	$t \in [0,\ 1.6]$
Control constraint	$-2 \le u \le 1$
Initial value	$x(0) = (1, 0)^T$
System of differential equations	$\begin{cases} \dot{x}_1 = \dfrac{1}{\cos x_1 + 2} + 3\sin x_2 + u \\ \dot{x}_2 = x_1 + x_2 + u \end{cases}$
Cost functional	$I(u) = -x_1(1.6) + \frac{1}{2}x_2(1.6)$

Optimization method and its parameters: hybrid multi-agent optimization method of interpolation search ($NP = 30$, $I_{\max} = 50$, $M_1 = 2$, $M_2 = 5$, $PRT = 0.01$, $n\text{step} = 5$, and $b_2 = 8$) and multi-agent optimization algorithm using linear regulators for agents motion control ($NP = 101$, $NMAX = 50$, $P_{\max} = 10$, $k_S = 0.1$, $k_\Lambda = 5$, and $h = 0.0001$).

Table 16.8 Results of solving task 3

Optimization method	Coordinates of points $(x_1(1.6),\ x_2(1.6))$	Switching point coordinate	The value of the functional I
Hybrid multi-agent optimization method of interpolation search	(3.45449, 12.87062)	1.25	−2.98082
Multi-agent optimization algorithm using linear regulators for agents motion control	(3.45949, 12.87562)	1.25	−2.97832
Known solution [15]	(3.46114, 12.884)	1.26	−2.98086

Table 16.9 Results of solving task 3

Optimization method	Coordinates of points $(x_1(1.6),\ x_2(1.6))$	Coefficients in expansion c_i	The value of the functional I
Hybrid multi-agent optimization method of interpolation search	(3.45449, 12.87062)	0.22, 1.15	−2.98082
Multi-agent optimization algorithm using linear regulators for agents motion control	(3.48523, 13.44901)	0.36, 0.99	−2.87218

The results of solving Task 3 by the search algorithm of optimal open-loop control using switching points are presented in Table 16.8.

Solving Task 3 by the search algorithm of optimal open-loop control using expansion in a system of basis functions. The best number of coefficients in expansion: $L = 2$.

Optimization method and its parameters: hybrid multi-agent optimization method of interpolation search ($NP = 30$, $I_{\max} = 50$, $M_1 = 2$, $M_2 = 5$, $PRT = 0.01$, $n\text{step} = 5$, and $b_2 = 8$) and multi-agent optimization algorithm using linear regulators for agents motion control ($NP = 101$, $NMAX = 50$, $P_{\max} = 10$, $k_s = 1$, $k_l = 5$, and $h = 0.0001$).

The results of solving Task 3 by the search algorithm of optimal open-loop control using expansion in a system of basis functions are presented in Table 16.9.

Graphs of optimal trajectories and controls are shown in Fig. 16.9.

16.4 Conclusions

Two new multi-agent algorithms are proposed to search for optimal open-loop control of one class of deterministic systems: the hybrid multi-agent method of interpolation search and the multi-agent method based on the use of linear regulators of

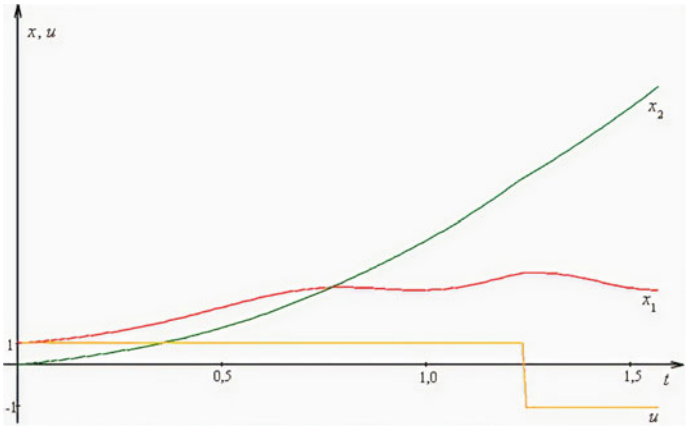


Fig. 16.9 Trajectories x_1 and x_2 and control u for task 3

agent movement control. In addition, algorithms for the search for optimal open-loop control on the basis of multi-agent methods are formed. In the first algorithm, it was proposed to represent the control in a relay form with a certain number of switching points. Second algorithm applies the spectral method and decomposes the control into a system of basis functions, for which cosine curves were used. Based on the described algorithms, software has been formed that allows finding the optimal open-loop control of nonlinear deterministic dynamical systems linear in bounded control. Three model examples were solved to analyze the effectiveness of the described algorithms for finding the optimal open-loop control. The solution found was compared with the known one. The result was close to optimal, from which we can conclude that the described algorithms successfully coped with the task.

References

1. Beheshti, Z., Shamsuddin, S.M.H.: A review of population-based meta-heuristic algorithms. Int. J. Adv. Soft Comput. Appl. **5**(1), 1–35 (2013)
2. Panovskiy, V.N., Panteleev, A.V.: Meta-heuristic interval methods of search of optimal in average control of nonlinear determinate systems with incomplete information about its parameters. J. Comput. Syst. Sci. Int. **56**(1), 52–63 (2017)
3. Panteleev, A.V., Pis'mennaya, V.A.: Application of a memetic algorithm for the optimal control of bunches of trajectories of nonlinear deterministic systems with incomplete feedback. J. Comput. Syst. Sci. Int. **57**(1), 25–36 (2018)
4. Panteleev, A., Metlitskaya, D.V.: An application of genetic algorithms with binary and real coding for approximate synthesis of suboptimal control in deterministic systems. Autom. Remote Control **72**(11), 2328–2338 (2011)
5. Panteleev, A.V., Metlitskaya, D.V.: Using the method of artificial immune systems to seek the suboptimal program control of deterministic systems. Autom. Remote Control **75**(11), 1922–1935 (2014)

6. Rere, L.M., Fanany, M.I., Arymurthy, A.: Metaheuristic algorithms for convolution neural network. Comput. Intell. Neuro Sci. **2016**, 1537325 (2016)
7. Panteleev, A., Karane, M.: Hybrid multi-agent optimization method of interpolation search. AIP Conf. Proc. **2181**, 020028 (2019)
8. Neumaier, A.: Personal page. https://www.mat.univie.ac.at/~neum/. Last accessed 2020/09/20
9. Mishra, S.K.: Some new test functions for global optimization and performance of repulsive particle swarm method. https://ssrn.com/abstract=926132. Last accessed 2020/09/20
10. Bacanin, N., Pelevic, B., Tuba, M.: Krill herd (KH) algorithm for portfolio optimization. In: International Conference on Mathematics and Computers in Business, Manufacturing and Tourism, pp. 39–44. Baltimore, USA (2013)
11. Gandomi, A.H., Alavi, A.H.: Krill herd: a new bio-inspired optimization algorithm. Commun. Nonlinear Sci. Numer. Simulat. **17**(5), 4831–4845 (2012)
12. Rybakov, K.A.: Modeling linear nonstationary stochastic systems by spectral method. Diff. Eqn. Control Process. (3), 98–128 (in Russian) (2020)
13. Rybakov, K.A.: Spectral method of analysis and optimal estimation in linear stochastic systems. Int. J. Model. Simul. Sci. Comput. **11**(3), 2050022 (2020)
14. Panteleev, A.V., Karane, M.M.S.: Multi-agent optimization algorithms for a single class of optimal deterministic control systems. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Computational Mechanics and Numerical Simulation. SIST, vol. 173, pp. 271–291. Springer, Singapore (2020)
15. Finkelstein, E.A.: Computational technologies of approximation of the reachable set of a controlled system: Dissertation for the Degree of Canada of Technology Sciences. Institute of System Dynamics and Control Theory, Irkutsk (in Russian) (2018)

Chapter 17
Modified Continuous-Time Particle Filter Algorithm Without Overflow Errors

Irina A. Kudryavtseva and Konstantin A. Rybakov

Abstract The modification for the continuous-time particle filter algorithm is offered. The developed modification that is based on the well-known strategy such as modeling trajectories to numerically solve stochastic differential equations provides the lack of overflow errors during the calculation of particle weights. To implement such an idea practically particle weights should be expressed in terms of logarithms with an additional customization of exponents. The effectiveness of the modified algorithm is demonstrated when solving the tracking problem to find coordinates and velocities of an aircraft executing a maneuver in the horizontal plane.

17.1 Introduction

The filtering problem for continuous-time stochastic systems given by two Stochastic Differential Equations (SDEs) describing an unobservable Markov random process and its measurements (two diffusion processes) is considered. The desired outcome is to estimate a system state vector from given measurements in accordance with some quality criterion [1–4]. Quality criteria can be chosen differently. For instance, one can take the following criteria: the Minimum Mean Squared Error (MMSE) criterion, the Maximum A Posteriori (MAP) criterion. Filtering problems arise in many fields among which we would like to highlight motion control and navigation data processing [5–16].

The goal of the paper is to develop the modification of the continuous-time particle filter algorithm [3]. This modification is rooted in the probability representation of

I. A. Kudryavtseva · K. A. Rybakov (✉)
Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, Moscow 125993, Russian Federation
e-mail: rkoffice@mail.ru

I. A. Kudryavtseva
e-mail: kudryavtseva.irina.a@gmail.com

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_17

the solution of the Duncan–Mortensen–Zakai (DMZ) equation [3, 17] for the unnormalized posterior probability density function of a random process whose trajectories are under estimation. Similar modifications can be developed for the particle filter algorithm based on the robust DMZ equation [14] and for the estimation of jump-diffusion processes as well [18].

Methods for solving the DMZ equation and the robust DMZ equation, among which the continuous-time particle filter algorithm can be particularly distinguished [4, 14], are given in [19–22].

According to the algorithm, it is supposed to simulate pairs including process trajectories themselves driven by a known state equation and associated weight functions that determine the importance of trajectories in an estimate needed to find. On the program implementation stage the overflow errors can appear because weight functions increase rapidly as exponential functions.

In the theory, the standard technique of the normalization of weights is a perfect instrument to gain the goal [3], but it fails in practice and the overflow errors can occur. To avoid appearing the mentioned errors it is offered to apply the procedure of taking logarithms, i.e. to switch from exponential functions to their exponents and as consequence from multiplying to summarizing, with additional customization of the exponents. Since relative values of weights but not their absolute values count for much in formulae for the optimal estimate, the customization does not affect the resulting estimate of unobserved process trajectories or the filtering problem solution in other words.

Note that the time discretization and the use of discrete-time particle filter algorithms [3, 23, 24] reduce the risk of overflow errors. However, if we need to apply the continuous-time particle filter algorithm, it should provide a lack of overflow errors.

The offered modification of the continuous-time particle filter algorithm is applied to solve the tracking problem to find coordinates and velocities of an aircraft executing a maneuver in the horizontal plane [6].

The remainder of this chapter is organized as follows. The optimal filtering problem is considered in Sect. 17.2. The known continuous-time particle filter algorithms and the offered modification that provides the lack of overflow errors are given in Sect. 17.3. Section 17.4 is devoted to the approbation of the new filtering algorithm. The chapter is summarized in Sect. 17.5.

17.2 Problem Formulation

The optimal filtering problem for continuous-time stochastic systems is to find an estimate of trajectories of an unobserved Markov random process $X(t)$ from given trajectories of an observed random process $Y(t)$ in accordance with the given quality criterion. The random processes $X(t)$ and $Y(t)$ satisfy the following system of Itô SDEs:

$$\mathrm{d}X(t) = f(t, X(t))\mathrm{d}t + \sigma(t, X(t))\mathrm{d}W(t), \quad X(t_0) = X_0, \tag{17.1}$$

$$\mathrm{d}Y(t) = c(t, X(t))\mathrm{d}t + \zeta(t)\mathrm{d}V(t), \quad Y(t_0) = Y_0 = 0, \tag{17.2}$$

where X is an n-dimensional state vector, Y is an m-dimensional measurement vector, $W(t)$ and $V(t)$ are the s-dimensional and d-dimensional independent standard Wiener processes, respectively, $f(t, x), \sigma(t, x), c(t, x), \zeta(t)$ are given vector-valued and matrix-valued functions with corresponding dimensions (the matrix $\zeta(t)\zeta^{\mathrm{T}}(t)$ must be nondegenerate), $t \in [t_0, T]$. Distribution of the vector X_0 is determined by the probability density function $\varphi_0(x)$. Equation 17.1 is the state equation while Eq. 17.2 is the measurement equation.

Coefficients in Eqs. 17.1 or 17.2 should satisfy the conditions on the existence and uniqueness of the solution of SDEs. According to [3], we assume that $f(t, x)$, $\sigma(t, x), c(t, x)$ are the Lipschitz functions with respect to x. Moreover, $\mathrm{E}|X_0|^2 < \infty$, where E denotes the mean.

We use MMSE criterion, this means $\hat{X}(t) = \mathrm{E}[X(t)|Y_0^t]$. Such an estimate provides the minimum value $\mathrm{E}|X(t) - \hat{X}(t)|^2$ for all $t \in [t_0, T]$.

Further, to form a sequence of relationships when solving the optimal filtering problem it is convenient to express Eq. 17.2 in the Langevin form:

$$Z(t) = \dot{Y}(t) = c(t, X(t)) + \zeta(t)N(t), \tag{17.3}$$

where $N(t)$ is a standard Gaussian white noise corresponding to the Wiener process $V(t)$. Since $Z(t)$ and $Y(t)$ are interchangeable in models of the considered type, the estimate of trajectories $X(t)$ can be found using measurements $Z(t)$, i.e. $\hat{X}(t) = \mathrm{E}[X(t)|Z_0^t]$.

17.3 Continuous-Time Particle Filter Based on the DMZ Equation

Simulating an ensemble of continuous-time stochastic system trajectories underlies the particle filter algorithm that will be given below. Moreover, it can be possible to construct an algorithm, where it is required to simulate trajectories of an auxiliary stochastic system whose mathematical model is formed with the system model defined by Eq. 17.1 and coefficients of Eqs. 17.2 or 17.3 [14]. Each trajectory is assigned a weight function whose values are calculated using given measurements. The estimation of a trajectory is carried out by applying the statistical treatment of results to calculate the weighted mean. Other statistical characteristics, for example, the mode, the posterior distribution function and the posterior probability density function can also be found from the same simulation results. Such a group of methods is named by particle filters. Behavior of a particle is determined by an ordered pair (trajectory, weight function). The continuous-time particle filter is described in more detail in [3].

As a rule, to simulate an ensemble of trajectories and corresponding weights the numerical methods for solving SDEs are employed. Both the simple method such as the Euler–Maruyama method having low order of convergence as a result that leads to low accuracy and methods of a higher order of convergence including those that possess the additional stability properties can be used [25–27]. If the filtering problem is needed to solve on manifolds [28–30], the specific modification of numerical methods given in [31, 32] should be applied.

The below relationships will be used for filtering algorithms:

$$X_{k+1} = F(t_k, X_k, h),$$
$$Y_{k+1} = C(t_k, X_k, Y_k, h),$$

where functions $F(t, X, h)$ and $C(t, X, Y, h)$ are determined by Eqs. 17.1–17.2 and the specific numerical method of solving SDEs. These functions involve $s \times 1$ and $d \times 1$ vectors whose components are independent Gaussian random variables with zero mean and unit variance to simulate increments of Wiener processes $W(t)$ and $V(t)$, respectively. They define discrete-time approximations for random processes $X(t)$ and $Y(t)$.

For instance, functions $F(t, X, h)$ and $C(t, X, Y, h)$ for the Euler–Maruyama are as follows:

$$F(t, X, h) = X + hf(t, X) + \sqrt{h}\sigma(t, X)\Delta W,$$
$$C(t, X, Y, h) = Y + hc(t, X) + \sqrt{h}\zeta(t)\Delta V,$$

where ΔW and ΔV are the s-dimensional and d-dimensional independent random vectors having a standard normal distribution. For Heun's method [33] we have:

$$F(t, X, h) = X + \frac{h}{2}\big(a(t, X) + a(t + h, X^*)\big) + \frac{\sqrt{h}}{2}\big(\sigma(t, X) + \sigma(t + h, X^*)\big)\Delta W,$$
$$X^* = X + hf(t, X) + \sqrt{h}\sigma(t, X)\Delta W, \quad a(t, x) = f(t, x) - \frac{1}{2}\sum_{l=1}^{s}\frac{\partial \sigma_{*l}(t, x)}{\partial x}\sigma_{*l}(t, x),$$

where $\sigma_{*l}(t, x)$ is the lth column of the matrix-valued function $\sigma(t, x)$. For the measurement equation, the use of the Euler–Maruyama method is sufficient because for the real filtering problem the measurements are given (measurements should not be simulated).

Heun's method provides acceptable computational accuracy, especially for SDEs with additive noise, while its implementation is not much more complicated as compared to the Euler–Maruyama method.

Simulating the continuous-time stochastic system 17.1, 17.2 and the algorithm of the continuous-time particle filter based on the DMZ equation are given below.

CONTINUOUS-TIME PARTICLE FILTER ALGORITHM (ALGORITHM 1)

1. Specify M, the number of auxiliary trajectories needed to simulate, h, the integration step. Draw the sample for initial state vectors X_0 and X_0^i from the given distribution with the probability density function $\varphi_0(x)$, $i = 1, 2, \ldots, M$. Set $k = 0$, $Y_0 = 0$, $\omega_0^i = 1$, $i = 1, 2, \ldots, M$.
2. Set

$$M_k = \sum_{i=1}^{M} \omega_k^i \quad (M_0 = M).$$

For the sample $\mathbb{X}_k = \{X_k^i\}_{i=1}^M$ with the set of associated weights $\mathbb{W}_k = \{\omega_k^i\}_{i=1}^M$ find the following statistics: the estimate of the state vector (the unbiased MMSE estimate)

$$\hat{X}_k = \frac{1}{M_k} \sum_{i=1}^{M} \omega_k^i X_k^i$$

and the estimate of the posterior covariance matrix

$$\hat{R}_k = \frac{1}{M_k} \sum_{i=1}^{M} \omega_k^i (X_k^i - \hat{X}_k)(X_k^i - \hat{X}_k)^{\mathrm{T}}.$$

Verify the condition $T - t_k = 0$. If it is met, then stop. Obtain a realization of the estimated state vector and the corresponding measurement vector at t_{k+1}:

$$X_{k+1} = F(t_k, X_k, h), \quad Y_{k+1} = C(t_k, X_k, Y_k, h), \quad Z_k = \frac{Y_{k+1} - Y_k}{h}.$$

3. Obtain a realization of the state vector at t_{k+1} and update the corresponding weight:

$$X_{k+1}^i = F(t_k, X_k^i, h),$$
$$\omega_{k+1}^i = \omega_k^i \mathrm{e}^{\mu(t_k, X_k^i, Z_k)h},$$

where

$$\mu(t, x, z) = c^{\mathrm{T}}(t, x) q(t) \left(z - \frac{1}{2} c(t, x) \right),$$
$$q(t) = \eta^{-1}(t), \quad \eta(t) = \zeta(t)\zeta^{\mathrm{T}}(t).$$

4. Verify conditions: if $i = M$ then set $t_{k+1} = t_k + h$, $k := k + 1$ and go to Step 2; if $i < M$ then set $i := i + 1$ and go to Step 3.

The algorithm given above is based on simulating the ensemble of stochastic system trajectories and weight functions that are trajectories of random processes $X(t)$ and $\omega(t)$. Trajectories of the random process $\omega(t)$ are drawn by definition [3]:

$$\omega(t) = \exp\left\{\int_{t_0}^{t} \mu(\tau, X(\tau), Z(\tau))\mathrm{d}\tau\right\}$$
$$= \exp\left\{\int_{t_0}^{t} c^{\mathrm{T}}(\tau, X(\tau))q(\tau)\mathrm{d}Y(\tau) - \frac{1}{2}\int_{t_0}^{t} c^{\mathrm{T}}(\tau, X(\tau))q(\tau)c(t, X(\tau))\mathrm{d}\tau\right\}.$$

In practice one can encounter underflow or overflow errors. Appearance of the underflow errors can be caused by the degeneracy when the weight $\omega(t)$ approaches zero while the overflow errors appear due to rapid increase of the weight. The simplest way to reduce the risk of their appearance is to normalize weights. For this reason, the weights should be redefined on Step 2 as follows:

$$\omega_k^i := \frac{\omega_k^i}{M_k}, \quad i = 1, 2, \ldots, M, \quad M_k = \sum_{i=1}^{M} \omega_k^i.$$

The proposed normalization technique does not affect the final solution because only the relative weights but not their absolute values contribute to the weighted mean calculated in Step 2. Modified algorithm containing the normalization of weights is given below.

CONTINUOUS-TIME PARTICLE FILTER ALGORITHM WITH NORMALIZATION OF WEIGHTS (ALGORITHM 2)

1. See Step 1 of Algorithm 1.
2. Set

$$M_k = \sum_{i=1}^{M} \omega_k^i \quad (M_0 = M)$$

and redefine weights:

$$\omega_k^i := \frac{\omega_k^i}{M_k}, \quad i = 1, 2, \ldots, M.$$

For the sample $\mathbb{X}_k = \{X_k^i\}_{i=1}^M$ with the set of associated weights $\mathbb{W}_k = \{\omega_k^i\}_{i=1}^M$ find the following statistics: the estimate of the state vector

$$\hat{X}_k = \sum_{i=1}^{M} \omega_k^i X_k^i$$

and the estimate of the posterior covariance matrix

$$\hat{R}_k = \sum_{i=1}^{M} \omega_k^i (X_k^i - \hat{X}_k)(X_k^i - \hat{X}_k)^{\mathrm{T}}.$$

Verify the condition $T - t_k = 0$. If it is met, then stop. Obtain a realization of the estimated state vector and the corresponding measurement vector at t_{k+1}:

$$X_{k+1} = F(t_k, X_k, h), \quad Y_{k+1} = C(t_k, X_k, Y_k, h), \quad Z_k = \frac{Y_{k+1} - Y_k}{h}.$$

3. See Step 3 of Algorithm 1.
4. See Step 4 of Algorithm 1.

The given standard normalization procedure is involved in both the discrete-time and continuous-time particle filters [3]. However, it might be insufficient when $\mu(t_k, X_k^i, Z_k)h$ is greater or lower than some threshold value. In this situation, the resampling procedure is also ineffective due to underflow or overflow errors. The threshold value is determined by the data type used for storage of the floating-point number. For example, when using the double-precision floating-point format (for storage of the floating-point number it is required 64 bit) this threshold value is about 706.893. Certainly, the value $\mu(t_k, X_k^i, Z_k)h$ can be lowered by the choice of the integration step but decreasing h proportionally increases the calculation time. And if it is required to solve the optimal filtering problem in real-time then such a way may be unrealizable. If the single-precision floating-point format is used (for storage of the floating-point number it is required 32 bit), then the threshold value is just 88.722. For the extended precision floating point format (for storage of floating-point numbers it is required 80 bit) the threshold value is 11,356.523, but this data type is used much more rarely in practice.

When overflow errors appear it is proposed to switch from the exponential function $\mathrm{e}^{\mu(t_k, X_k^i, Z_k)h}$ to the expression involving the exponent $\mu(t_k, X_k^i, Z_k)h$ and the quantity that provides the correctness of this procedure. In other words, it provides the exponential function calculation without overflow errors. For that, instead of the expression

$$\omega_{k+1}^i = \omega_k^i \mathrm{e}^{\mu(t_k, X_k^i, Z_k)h}$$

it is used the relation

$$\omega_{k+1}^i = \exp\{\ln \omega_k^i + \mu(t_k, X_k^i, Z_k)h - \gamma_k\},$$

where γ_k is determined sequentially for each time moment t_k by the formula:

$$\gamma_k = \max_{i=1,2,\ldots,M} \{\ln \omega_k^i + \mu(t_k, X_k^i, Z_k)h\}.$$

The described technique, as well as, the standard normalization does not affect the final result because all weights are multiplied by the same factor $\mathrm{e}^{-\gamma_k}$.

CONTINUOUS-TIME PARTICLE FILTER ALGORITHM BASED ON TAKING LOGARITHMS (ALGORITHM 3)

1. See Step 1 of Algorithm 1.
2. See Step 2 of Algorithm 1.
3. Obtain a realization of the state vector at t_{k+1} and update the corresponding weight:

$$X_{k+1}^i = F(t_k, X_k^i, h),$$
$$\omega_{k+1}^i = \exp\{\ln \omega_k^i + \mu(t_k, X_k^i, Z_k)h - \gamma_k\},$$

where

$$\gamma_k = \max_{i=1,2,\ldots,M} \{\ln \omega_k^i + \mu(t_k, X_k^i, Z_k)h\}.$$

4. See Step 4 of Algorithm 1.

Note that these algorithms include modeling the trajectory of a random process $X(t)$ as well as its measurements $Y(t)$ and $Z(t)$ on Step 2 (and also modeling the initial state vector X_0 on Step 1). It is needed for the simulation purpose only. In the real filtering problem, the random process $X(t)$ is unobservable, and measurements $Y(t)$ or $Z(t)$ are given.

In Algorithm 3, at least one particle has weight 1 for all $k = 0, 1, \ldots, N$, and this weight is maximal in the set $\mathbb{W}_k$, therefore, $M_k \neq 0$. Thus, this algorithm provides not only the lack of overflow errors but also the lack of underflow errors, when all weights become zero in one step. It is important to emphasize that the proposed modification of the continuous-time particle filter does not exclude the resampling procedure, but complements it. In Algorithms 1–3, the resampling procedure is not described only for briefness.

All algorithms mentioned above are implemented in Mathcad computer algebra system and grouped in the software modules. This software permits to solve estimating problems (filtering, smoothing, and prediction problems) for multidimensional continuous-time stochastic systems. In addition to the described algorithms, the various numerical methods for solving SDEs such as Euler–Maruyama method, Runge–Kutta type methods, Milstein's method, Kuznetsov's method, Platen and Rosenbrock type methods have been implemented in the developed software. Moreover, MAP estimator [14, 34] can also be used for solving the optimal filtering

problem. The developed software has been tested on the tracking problem described in the next section.

17.4 Simulations

The modified continuous-time particle filter algorithm is applied to solve the tracking problem to find coordinates and velocities of an aircraft executing a maneuver in the horizontal plane [6].

Suppose $X = [\varepsilon, \dot{\varepsilon}, \eta, \dot{\eta}, \zeta, \dot{\zeta}, \omega]^{\mathrm{T}}$ is a state vector, where ε, η, ζ are coordinates of the aircraft in the Cartesian coordinate system, $\dot{\varepsilon}$, $\dot{\eta}$, $\dot{\zeta}$ are the corresponding velocities, ω is an angular velocity of the aircraft, X is the 7×1 vector.

The motion model is

$$\mathrm{d}\begin{pmatrix} \varepsilon(t) \\ \dot{\varepsilon}(t) \\ \eta(t) \\ \dot{\eta}(t) \\ \zeta(t) \\ \dot{\zeta}(t) \\ \omega(t) \end{pmatrix} = \begin{pmatrix} \dot{\varepsilon}(t) \\ -\omega(t)\dot{\eta}(t) \\ \dot{\eta}(t) \\ \omega(t)\dot{\varepsilon}(t) \\ \dot{\zeta}(t) \\ 0 \\ 0 \end{pmatrix} \mathrm{d}t + \begin{pmatrix} 0 & 0 & 0 & 0 \\ \sigma_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \sigma_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_1 & 0 \\ 0 & 0 & 0 & \sigma_2 \end{pmatrix} \mathrm{d}W(t),$$

where $\sigma_1 = \sqrt{0.2}$, $\sigma_2 = 0.007$; $W(t)$ is a 4×1 vector; σ_1 and σ_2 describe influence of unpredictable factors on the aircraft motion such as turbulence, wind gusts, etc. The initial state vector

$$X_0 = [2650\ \text{m}, \quad 150\ \text{m/s}, \quad 1000\ \text{m}, \quad 0\ \text{m/s}, \quad 200\ \text{m}, \quad 0\ \text{m/s}, \quad 6^{\circ}/\text{s}]^{\mathrm{T}}$$

is non-random.

For the chosen value of the angular velocity, the aircraft alters its heading toward East at $t = 15$ s, toward South at $t = 30$ s, toward West at $t = 45$ s, and returns to North at $t = 60$ s with insufficient deviations caused by random fluctuations.

The measurement vector $Z = [r, \theta, \phi]^{\mathrm{T}}$ satisfies the equation:

$$\begin{pmatrix} r(t) \\ \theta(t) \\ \phi(t) \end{pmatrix} = \begin{pmatrix} \sqrt{\varepsilon^2(t) + \eta^2(t) + \zeta^2(t)} \\ \arctan \frac{\eta(t)}{\varepsilon(t)} \\ \arctan \frac{\zeta(t)}{\sqrt{\varepsilon^2(t)+\eta^2(t)}} \end{pmatrix} + \begin{pmatrix} \sigma_r & 0 & 0 \\ 0 & \sigma_\theta & 0 \\ 0 & 0 & \sigma_\phi \end{pmatrix} N(t),$$

where $\sigma_r = 50$ m, $\sigma_\theta = 0.1^{\circ}$, $\sigma_\phi = 0.1^{\circ}$. The vector Z, as well as, $V(t)$ and $N(t)$ are 3×1 vectors, r is the distance from the origin, where radar is located, to the aircraft, θ is the azimuth, ϕ is the elevation angle.

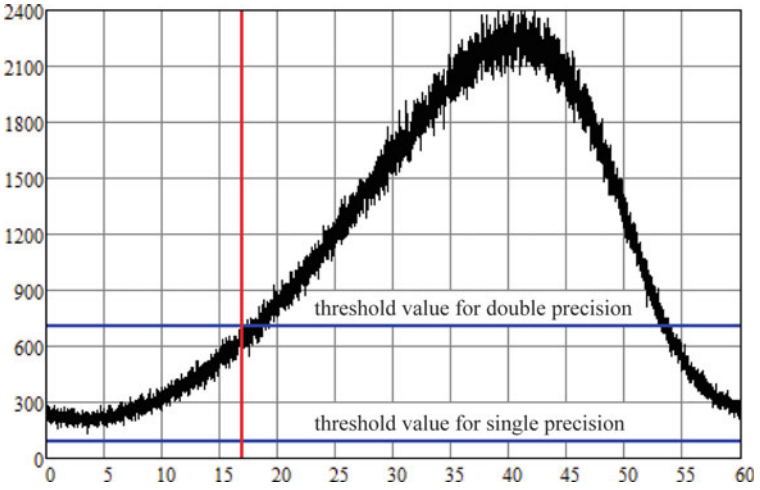


Fig. 17.1 The graph of $\mu(t_k, X_k, Z_k)h$ for sample trajectories of random processes $X(t)$ and $Z(t)$

MMSE criterion is chosen as a quality criterion. In the considered model, several corrections in comparison with the model in [6] have been made.

Firstly, the matrix $\sigma(t, x)$ has been restricted in the motion model. In [6], the matrix $\sigma(t, x)$ is the 7×7 matrix $\sigma(t, x) = \mathrm{diag}(0, \sigma_1, 0, \sigma_1, 0, \sigma_1, \sigma_2)$. As it is distinctly seen a number of its zero elements are superfluous that leads to extra calculations related to drawing a pseudorandom sequence when simulating aircraft trajectories. In the modified model the matrix $\sigma(t, x)$ has a reduced dimension 7×4 and does not contain zero columns. Secondly, the initial condition has been changed so that aircraft trajectories do not cross the plane $\varepsilon = 0$.

Figure 17.1 shows the graph $\mu(t_k, X_k, Z_k)h$ as a function of the time moments t_k for trajectories of random processes $X(t)$ and $Z(t)$ that are obtained by Heun's method [33] with the integration step $h = 0.01$. The threshold value 706.893 indicating a level, where the overflow error appears, is also marked in Fig. 17.1. This overflow error occurs at $t = 17$s. It should be emphasized that the overflow error appears not only in the case of implementation in Mathcad but in any other application supporting the double-precision floating-point format. The single-precision floating-point format is unusable here (see Fig. 17.1).

The standard normalization (see Algorithm 2) is ineffective in the latter case as the overflow error appears in one step at a time but it is not caused by the gradual increase of the weight coefficient.

Several ways of grappling with the discussed question can be offered. The first way is to reduce the integration step. For example, in considered simulations, the integration step should be decreased even to 0.003. The second one is to carry out calculations with the extended precision floating point format. Finally, the last way is to apply Algorithm 3.

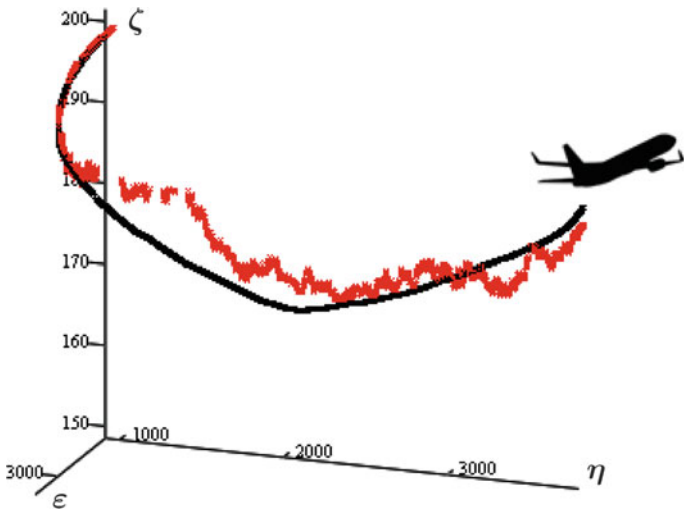


Fig. 17.2 The aircraft sample trajectory and its optimal estimate by Algorithm 3

The aircraft sample trajectory and its optimal estimate by Algorithm 3 are presented in Fig. 17.2. These numerical results correspond to the integration step $h = 0.01$, the sample size $M = 10{,}000$, and $T = 30$ s.

The drawback of the first way is undoubtedly growth of the calculation time. Actually, in spite of the disappearance of the overflow error at each time step, it might appear because of the increase of weights. The second variant has restrictions connected with the fact that the extended precision floating point format must be supported by the processor providing all needed calculations and the application development environment. Moreover, computer memory usage increases by 25%, as well as, the calculation time. Following the third way, the calculation time increases only because of finding the maximal element of the array (see Step 3 in Algorithm 3), but this way provides to eliminate overflow errors that are illustrated above.

17.5 Conclusions

In the chapter, the modification for the continuous-time particle filter algorithm has been offered. This modification is based on the well-known strategy such as modeling trajectories to numerically solve SDEs; it provides the lack of overflow errors during the calculation of particle weights. To implement such an idea practically particle weights have been expressed in terms of logarithms with additional customization of exponents. The effectiveness of the modified algorithm has been demonstrated when solving the tracking problem to find coordinates and velocities of an aircraft executing a maneuver in the horizontal plane.

References

1. Jazwinski, A.H.: Stochastic Processes and Filtering Theory. Academic Press, New York (1970)
2. Bar-Shalom, Y., Li, X.R., Kirubarajan, T.: Estimation with Applications to Tracking and Navigation. Wiley, New York (2001)
3. Bain, A., Crisan, D.: Fundamentals of Stochastic Filtering. Springer, New York (2009)
4. Crisan, D.: The stochastic filtering problem: A brief historical account. Appl. Probab. Index **51**(A), 13–22 (2014)
5. Gustafsson, F., Gunnarsson, F., Bergman, N., Forssell, U., Jansson, J., Karlsson, R., Nordlund, P.-J.: Particle filters for positioning, navigation and tracking. IEEE Trans. Signal Process. **50**(2), 425–437 (2002)
6. Arasaratnam, I., Haykin, S., Hurd, T.R.: Cubature Kalman filtering for continuous-discrete systems: theory and simulations. IEEE Trans. Signal Process. **58**(10), 4977–4993 (2010)
7. Chari, V., Jawahar, C.: Multiple plane tracking using unscented Kalman filter. In: Proceedings of 50th IEEE Conference on Decision and Control and European Control Conference, pp. 2914–2919 (2011)
8. Jeon, D., Eun, Y., Bang, H., Yeom, C.: Nonlinear aircraft tracking filter utilizing a point mass flight dynamics model. J. Aerosp. Eng. **227**(11), 1795–1810 (2012)
9. Song, H., Shin, V., Jeon, M.: Mobile node localization using fusion prediction-based interacting multiple model in cricket sensor network. IEEE Trans. Ind. Electron. **59**(11), 4349–4359 (2012)
10. Cook, B., Arnett, T., Macmann, O., Kumar, M.: Real-time radar-based tracking and state estimation of multiple non-conformant aircraft. In: Proceedings of 55th AIAA Aerospace Sciences Meeting (2017)
11. Zorina, O.A., Izmailov, E.A., Kukhtevich, S.E., Portnov, B.I., Fomichev, A.V., Vavilova, N.B., Golovan, A.A., Papusha, I.A., Parusnikov, N.A.: Enhancement of INS/GNSS integration capabilities for aviation-related applications. Gyroscopy Navig. **8**(4), 248–258 (2017)
12. Rudenko, E.A.: Autonomous path estimation for a descent vehicle using recursive Gaussian filters. J. Comput. Sys. Sc. Int. **57**(5), 695–712 (2018)
13. Rudenko, E.A.: Optimal recurrent nonlinear filter of a large order for jump diffusion Markov signals. J. Comput. Sys. Sc. Int. **59**(1), 49–62 (2020)
14. Chugai, K.N., Kosachev, I.M., Rybakov, K.A.: Approximate Filtering Methods in Continuous-Time Stochastic Systems. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Computational Mechanics and Numerical Simulation. Smart Innovation, Systems and Technologies, vol. 173, pp. 351–371. Springer, Singapore (2020)
15. Tupysev, V.A., Litvinenko, Yu.A.: Application of polynomial-type filters to integrated navigation systems with modular architecture. In: Proceedings of 26th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS) (2019)
16. Stepanov, O.A., Vasiliev, V.A., Toropov, A.B., Loparev, A.V., Basin, M.V.: Efficiency analysis of a filtering algorithm for discrete-time linear stochastic systems with polynomial measurements. J. Franklin Inst. **356**(10), 5573–5591 (2019)
17. Hazewinkel, M.: Lectures on linear and nonlinear filtering. In: Schiehlen, W.O., Wedig, W. (eds.) Analysis and Estimation of Stochastic Mechanical Systems. International Centre for Mechanical Sciences (Courses and Lectures), vol. 303, pp. 103–136. Springer, Vienna (1988)
18. Averina, T.A., Rybakov, K.A.: Using maximum cross section method for filtering jump-diffusion random processes. Rus. J. Numer. Anal. Math. Modelling **35**(2), 55–67 (2020)
19. Lototsky, S., Mikulevicius, R., Rozovskii, B.L.: Nonlinear filtering revisited: a spectral approach. SIAM J. Control Optim. **35**(2), 435–461 (1997)
20. Ahmed, N.U., Radaideh, S.M.: A powerful numerical technique solving Zakai equation for nonlinear filtering. Dyn. Control **7**(3), 293–308 (1997)
21. Gobet, E., Pagès, G., Pham, H., Printemps, J.: Discretization and simulation of Zakai equation. SIAM J. Numer. Anal. **44**(6), 2505–2538 (2006)
22. Luo, X., Yau, S.S.-T.: Complete real time solution of the general nonlinear filtering problem without memory. IEEE Trans. Autom. Control **58**(10), 2563–2578 (2013)

23. Doucet, A., de Freitas, N., Gordon, N. (eds.): Sequential Monte Carlo Methods in Practice. Springer, New York (2001)
24. Cappé, O., Godsill, S.J., Moulines, E.: An overview of existing methods and recent advances in sequential Monte Carlo. Proc. IEEE **95**(5), 899–924 (2007)
25. Kloeden, P.E., Platen, E.: Numerical Solution of Stochastic Differential Equations. Springer-Verlag, Berlin (1995)
26. Artemiev, S.S., Averina, T.A.: Numerical Analysis of Systems of Ordinary and Stochastic Differential Equations. VSP, Utrecht (1997)
27. Kuznetsov, D.F.: Expansion of iterated Stratonovich stochastic integrals based on generalized multiple Fourier series. Ufa Math. J. **11**(4), 49–77 (2019)
28. Koval, M.C., Klingensmith, M., Srinivasa, S.S., Pollard, N.S., Kaess, M.: The manifold particle filter for state estimation on high-dimensional implicit manifolds. In: Proceedings of IEEE International Conference on Robotics and Automation (ICRA), pp. 4673–4680 (2017)
29. Marjanovic, G., Solo, V.: An engineer's guide to particle filtering on the Stiefel manifold. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2017)
30. Rybakov, K.A.: On a class of filtering problems on manifolds. Informatika i ee Primeneniya **13**(1), 16–24 (in Russian) (2019)
31. Averina, T.A., Karachanskaya, E.V., Rybakov, K.A.: Statistical modeling of random processes with invariants. In: Proceedings of IEEE International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON), pp. 34–37 (2017)
32. Averina, T.A., Rybakov, K.A.: A modification of numerical methods for stochastic differential equations with first integrals. Numer. Anal. Appl. **12**(3), 203–218 (2019)
33. Kloeden, P.E., Pearson, R.A.: The numerical solution of stochastic differential equations. J. Aust. Math. Soc. B **20**, 8–12 (1977)
34. Chugai, K., Kosachev, I., Rybakov, K.: Approximate MMSE and MAP estimation using continuous-time particle filter. AIP Conf. Proc. **2181**, 020001 (2019)

Chapter 18
Incomplete Pairwise Comparisons Method for Estimating the Impact Criteria for Hub Airports Network Optimization

Nataliya M. Kuzmina and Alexandra N. Ridley

Abstract This chapter proposes a solution to the problem of determining the contribution of airport rating criteria for assessing the integral risk of modernization. The purpose of modernization is to increase the throughput of the Moscow aviation hub. To solve this problem, the experts formed a list of criteria. To determine their contribution to the overall risk assessment, the method of pairwise comparisons was applied. However, due to the fact that the criteria being evaluated relate to different areas, and experts from different areas were involved, an interval system of evaluating alternatives based on Saaty's scale was used. The results of expert evaluations were combined into a common matrix of interval evaluations of alternatives. To increase the consistency of the matrix, some elements were deleted. The method proposed to solve the problem allows us to obtain the alternatives' weights for incomplete pairwise comparisons matrices of large dimension, as well as, alternative estimates in interval form, which is illustrated by an example. This method differs from most existing methods for solving the problem of incomplete pairwise comparisons by the ability to process incomplete pairwise comparisons matrices without restoring missing data. It can be applied to solve other decision problems, where most of the known methods based on the pairwise comparisons method are not applicable to solve a problem.

N. M. Kuzmina
The Moscow State Technical University of Civil Aviation, 20, Kronshtadtsky blvd, Moscow 125993, Russian Federation
e-mail: n.kuzmina@mstuca.aero

A. N. Ridley (✉)
Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, Moscow 125993, Russian Federation
e-mail: alexandra.ridley@yandex.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_18

18.1 Introduction

Currently, the share of passenger traffic at Moscow Aviation Hub (MAH) airports is about 75% of the passenger traffic of all airports in Russia [1]. Serving such a share of passenger traffic requires the use of significant throughput capacities. Currently, the Moscow aviation hub includes three airports: Domodedovo, Sheremetyevo, and Vnukovo. The total passenger flow currently exceeds 115 million passengers per year (2019). MAH is the third busiest aviation hub in Europe (after London and Paris) and is among the ten most loaded in the world. There are two possible ways to solve the bandwidth problem. The first way is to modernize the external ground, airfield, and terminal infrastructure of existing airports [2], the Unified Air Traffic Management System (EU ATM), but this method is ineffective, since it will not provide service for the passenger flow projected for 2030, equal to 180 million passengers [3]. The second, most effective way is to create a new hub airport. Obviously, building a new airport from scratch is inefficient, it is better to use the existing airports near Moscow and modernize in accordance with the necessary requirements. In this regard, the task arose of finding the most reliable airport candidate for modernization in order to unload MAH. The attractiveness of the airport for the role of a hub is determined by the following factors:

- Geopolitical position (capital, the center of the economic region).
- Development of its infrastructure (airfield, passenger and cargo terminal, transport communications, Air Traffic Control (ATC) system.
- Capacity and development of the domestic and international transportation market.
- Level of passenger and cargo services at the airport.

Along with this, there are several criteria that determine the willingness of an airport to fulfill the functions of a hub:

- Ability to organize the required number of connections and the absence of restrictions for their growth.
- Ability to organize connecting flights by organizing connecting waves and clearly following the schedule.
- Possibility of developing an airport because there is no prospect without it.
- Presence of a hub-forming airline or airline alliance.
- Airport development opportunity.

The rest of the chapter is follows. Section 18.2 contains the statement of the problem including the task restrictions and using the model and list of criteria that may influence airport rating. Section 18.3 contains a description of the problem of criteria ranking based on incomplete, interval-specified pairwise comparisons. Section 18.4 concludes the chapter.

18.2 General Statement of the Problem

To select a hub airport, it was decided to rank airports within a radius of 300 km [4] in the terms of the integral risk [5] of airport modernization. This approach was chosen because the cost of modernizing airports should be determined after the preliminary selection of candidate airports for the feasibility of any investment. It is obvious, for example, that airports outside the 300 km zone will not unload MAH due to the fact that it is more profitable for passengers to fly by plane than to use an alternative mode of transport such as a train, intercity bus, taxi, and others. Integral risk is determined on the basis of a list of criteria and particular risks for each of the criteria for each airport in question. In this regard, the task is extremely important for obtaining the final result.

First, experts were involved to form lists of criteria that hypothetically may influence the magnitude of the defined risk. As a result, a list of criteria is formulated in Table 18.1.

Despite a large number of criteria, due to their diversity and the requirement to take into account the opinions of experts from different fields, the paired comparisons method [6] was chosen to evaluate the criteria, allowing experts to set the estimates in a form of intervals while skipping pairs of objects that seem difficult to compare. The method of incomplete pair comparisons [7] chosen to solve the problem was applied to solve complex, multi-criteria decision-making problems with a large number of criteria, both for choosing infrastructure facilities, assessing their safety, and for solving civil aviation problems [8].

Table 18.1 Airport rating criteria

No.	Criteria	Acronyms
1	Optimal distance from the center of Moscow	ODC
2	Airport capacity	ACC
3	Quality and quantity of runways	QQRC
4	Aerodrome infrastructure	ADIC
5	Airport infrastructure	APIC
6	Land resources	LRC
7	Infrastructure of other transport	ITC
8	Cargo terminal quality	CTC
9	International status	ISC
10	Co-location	CLC
11	Ownership form	OSC

18.3 Evaluation of the Contribution of Criteria Based on the Pairwise Comparisons Method

The definitions of the incomplete pairwise comparison solutions without and with interval alternative preference ratings are provided in Sects. 18.3.1 and 18.3.2, respectively. Section 18.3.3 contains a method of processing expert data for increasing matrix consistency. Section 18.3.4 is the main subsection. It contains the description of the algorithm to obtain a solution to the problem and results of its execution—solution of the criteria ranking problem. Section 18.3.5 contains a basic analysis of the results.

18.3.1 Incomplete Pairwise Comparisons Method

Let $\{O_1, O_2, \ldots, O_N\}$ be a set of alternatives, where N is the object count. $S = \left(s_{ij}\right), 1 \le i, j \le N$ is the pairwise comparisons matrix, where alternatives preference relations using Saaty's scale. Some pairs may not be rated and it will be marked "NA" in pairwise comparisons matrix. Note that all relations are inverse symmetric ($s_{ij} = s_{ji}^{-1}$) or both marked NA [7–9].

Let's construct a directed weighted graph $G_s := (V, E)$, where $V = \{O_1, O_2, \ldots, O_N\}$ are the vertices of this graph, $E = \left\{\langle i, j\rangle : s_{ij} \ne NA, s_{ij} > s_{ji}\right\}$ are the edges and weights correspond to values from s_{ij}.

Select a connected subgraph G_s^* from graph G_s: delete edges from G_s, which mostly disrupt transitivity $s_{ik} = s_{ij}s_{jk}$. Thus, for all edges $\langle i, j\rangle$ in G_s^* will be $s_{ij} = w_i/w_j$. This will uniquely identify the alternative weights vector $W = (w_i)$ is the solution of the incomplete pairwise comparisons problem.

18.3.2 Incomplete Pairwise Comparisons Method with Interval Alternative Preference Ratings

Let $\left\{C^1, C^2, \ldots, \mathrm{C^M}\right\}$ be the survey results obtained from M experts. They are presented as incomplete pairwise comparisons matrices with interval estimates of N alternatives. That is all elements are determined by Eq. 18.1 or as NA.

$$C^m = \left(c_{ij}^m\right), c_{ij}^m = \left[bottom_{ij}^m, top_{ij}^m\right] \quad 1 \le m \le M \quad 1 \le i, j \le N \tag{18.1}$$

Due to the principle of inverse symmetry of pairwise comparisons matrices, the following equality will be true:

$$bottom_{ij}^m \cdot top_{ji}^m = 1. \tag{18.2}$$

Let $C = (c_{ij})$ be the combined results of survey formed from set $\{C^1, C^2, \ldots, C^M\}$. Values are presented as intervals $c_{ij} = [b_{ij}, t_{ij}]$, where b_{ij} are the values of lower bound and t_{ij} are the values of upper bound. Bounds are defined according to Eq. 18.3.

$$b_{ij} = \left(\prod_{k=1}^{M} bottom_{ij}^{k}\right)^{\frac{1}{M}} \quad t_{ij} = \left(\prod_{k=1}^{M} top_{ij}^{k}\right)^{\frac{1}{M}} \tag{18.3}$$

The solution to incomplete pairwise comparisons problem with interval alternatives' ratings is the solution to incomplete pairwise comparisons problem, which corresponds to Eq. 18.4.

$$W : b_{ij} \leq w_i / w_j \leq t_{ij} \tag{18.4}$$

It is easy to see that in most cases there is infinite number of solutions. That is why we determine two supporting solutions: upper (or top) solution and lower (or bottom) solution. Top solution is the solution to the incomplete pairwise comparisons problem for matrix $T = (t_{ij})$. Bottom solution is built by moving top solution to lower bound. Middle solution builds based on top solution and bottom solution. Algorithm of getting this solution will be described in the following sections.

18.3.3 Processing Expert Data

Using expert estimates in pairwise comparisons matrices form (like was described in Eq. 18.1) we got two matrices of coefficients: upper bound T and lower bound B. Values of these matrices are presented in Tables 18.2 and 18.3, respectively.

Table 18.2 Upper bound coefficients

	ODC	ACC	QQRC	ADIC	APIC	LRC	ITC	CTC	ISC	CLC	OSC
ODC	1.0000	3.9360	1.4142	1.8612	2.2134	1.8612	1.3161	7.4833	9.0000	9.0000	9.0000
ACC	0.3433	1.0000	0.4083	0.4518	0.5000	0.4083	0.3689	2.2134	2.9130	3.7606	4.3559
QQRC	1.0000	3.4641	1.0000	1.1892	1.6818	1.4142	1.1892	6.7354	7.4539	9.0000	9.0000
ADIC	0.8409	2.7108	1.0000	1.0000	1.1892	1.0000	1.0000	5.4772	7.9686	8.2067	9.0000
APIC	0.7071	2.2134	0.8409	1.0000	1.0000	1.0000	0.7071	4.7287	6.0000	7.4833	9.0000
LRC	1.0000	3.0000	1.1892	1.1892	2.2134	1.0000	1.0000	6.2357	8.4519	7.7373	9.0000
ITC	1.1892	3.4641	1.3161	1.4142	2.0000	1.1892	1.0000	7.2376	9.0000	9.0000	9.0000
CTC	0.1543	0.5000	0.1757	0.2115	0.2541	0.1931	0.1615	1.0000	1.5651	2.0000	2.2795
ISC	0.1179	0.4518	0.1543	0.1964	0.1931	0.1679	0.1361	1.1892	1.0000	1.4142	1.8612
CLC	0.1111	0.4205	0.1183	0.1436	0.1708	0.1382	0.1144	1.0000	1.0000	1.0000	1.1892
OSC	0.1111	0.3433	0.1111	0.1309	0.1183	0.1179	0.1111	0.9036	1.0000	1.1892	1.0000

Table 18.3 Lower bound coefficients

	ODC	ACC	QQRC	ADIC	APIC	LRC	ITC	CTC	ISC	CLC	OSC
ODC	1.0000	2.9130	1.0000	1.1892	1.4142	1.0000	0.8409	6.4807	8.4853	9.0000	9.0000
ACC	0.2541	1.0000	0.2887	0.3689	0.4518	0.3333	0.2887	2.0000	2.2134	2.3784	2.9130
QQRC	0.7071	2.4495	1.0000	1.0000	1.1892	0.8409	0.7598	5.6924	6.4807	8.4520	9.0000
ADIC	0.5373	2.2134	0.8409	1.0000	1.0000	0.8409	0.7071	4.7287	5.0915	6.9640	7.6372
APIC	0.4518	2.0000	0.5946	0.8409	1.0000	0.4518	0.5000	3.9360	5.1800	5.8560	8.4519
LRC	0.5373	2.4495	0.7071	1.0000	1.0000	1.0000	0.8409	5.1800	5.9579	7.2376	8.4853
ITC	0.7598	2.7108	0.8409	1.0000	1.4142	1.0000	1.0000	6.1920	7.3485	8.7389	9.0000
CTC	0.1336	0.4518	0.1485	0.1826	0.2115	0.1604	0.1382	1.0000	0.8409	1.0000	1.1067
ISC	0.1111	0.3433	0.1342	0.1255	0.1667	0.1183	0.1111	0.6389	1.0000	1.0000	1.0000
CLC	0.1111	0.2659	0.1111	0.1219	0.1336	0.1292	0.1111	0.5000	0.7071	1.0000	0.8409
OSC	0.1111	0.2296	0.1111	0.1111	0.1111	0.1111	0.1111	0.4387	0.5373	0.8409	1.0000

Pairwise comparisons matrices of dimensions more than 5-6 are initially highly controversial. We can increase consistency of these matrices by deleting preferences (mark them NA), which introduces inconsistency into the matrix because of scale limits and some features. First, we delete all preferences, where the top and bottom values are in (8, 9] both or in [1/9, 1/8). They are bad because of situations where two different objects are substantially, absolute more important than the third. But marks 9 make them more the same than different. These values are marked by red color in Tables 18.2, 18.3 and 18.4. Second, we delete all preferences, where the top and bottom values are both in (1/2, 2). Neutral preferences are bad because they spoil not neutral marks. These values are marked by the yellow color in Tables 18.2, 18.3 and 18.4. Also, we can see that pairwise comparisons matrices are inverse symmetrical. Therefore, we can delete half of the preferences: for example, these values where the top value is less than 1. These values are marked by green color in Tables 18.2, 18.3 and 18.4. Diagonal values are marked by gray color in Tables 18.2, 18.3 and 18.4 for

Table 18.4 Adjacency matrix of preferences graph after unlinking

	ODC	ACC	QQRC	ADIC	APIC	LRC	ITC	CTC	ISC	CLC	OSC
ODC	0	1	0	0	1	0	0	1	0	0	0
ACC	0	0	0	0	0	0	0	1	1	1	1
QQRC	0	1	0	0	0	0	0	1	1	0	0
ADIC	0	1	0	0	0	0	0	1	1	1	1
APIC	0	1	0	0	0	0	0	1	1	1	0
LRC	0	1	0	0	1	0	0	1	1	1	0
ITC	0	1	0	0	1	0	0	1	1	0	0
CTC	0	0	0	0	0	0	0	0	0	1	1
ISC	0	0	0	0	0	0	0	0	0	0	0
CLC	0	0	0	0	0	0	0	0	0	0	0
OSC	0	0	0	0	0	0	0	0	0	0	0

better orientation. Finally, we got upper and lower bounds of the expert's preferences. Their presence or absence can be defined with graph G, whose adjacency matrix is represented in Table 18.4.

18.3.4 Weights Calculating

Preparatory stage. At this stage, we go to the logarithmic scale. All matrices, vectors, and constants except matrix of preferences graph are calculating in logarithmic scale to increase accuracy and make it easier. Next, we denote logarithmic equivalents of linear quantities with the dash above. Logarithmic bounds are calculated by Eq. 18.5.

$$\bar{b}_{ij} = \ln b_{ij} \bar{t}_{ij} = \ln t_{ij} \tag{18.5}$$

For calculations, we introduce two matrices: the proximity matrix to upper bound $\bar{E} = (\bar{e}_{ij})$ and the proximity matrix to lower bound $\bar{R} = (\bar{r}_{ij})$. Their values will vary depending on the values of alternatives' weight vector $\bar{W} = (\bar{w}_i)$ according to the rules provided by Eq. 18.6.

$$\begin{cases} \bar{e}_{ij} = g_{ij} \cdot (\bar{t}_{ij} - \bar{w}_i + \bar{w}_j) \\ \bar{r}_{ij} = g_{ij} \cdot (-\bar{b}_{ij} + \bar{w}_i - \bar{w}_j) \end{cases} \tag{18.6}$$

First, when $\bar{w}_i = 0$, we obtain matrices $\bar{E} = \bar{T}$ and $\bar{R} = -\bar{B}$. Solutions to the problem will provide $\bar{e}_{ij} \geq 0$ и $\bar{r}_{ij} \geq 0$, i.e. corresponding incomplete pairwise comparisons matrix will be between upper and lower bounds.

Upper solution construction. Until there is a solution where $\max\limits_{i,j} |\bar{e}_{ij}| = 0$, it is necessary to remove the edges that introduce the greatest inconsistency [9]. The search for the required edge is carried out by changing the value $\bar{w}_i$ until there is no way to change weights to make $\max\limits_{i,j} |\bar{e}_{ij}|$ lower. For example, Table 18.5 illustrates alternatives' weights vector $\bar{w}_i$ and upper bound proximity matrix $\bar{E}$ corresponding to weight vector and upper bound values, which are presented in Table 18.2. We colored table cells corresponding to adjacency graph G.

In this matrix $\max\limits_{i,j} |\bar{e}_{ij}| = 0.2537$ (these values in Table 18.5 are marked in bold). This maximum is impossible to decrease. In Table 18.6, there are 3 nonreducible maximums: $\bar{e}_{49}$, $\bar{e}_{54}$, and $\bar{e}_{59}$. Let's try to increase $\bar{w}_4$ by Δw to decrease values in row 4. Updated value is $\bar{e}_{49} = 0.2537 - \Delta w$. But increasing $\bar{w}_4$ makes values in column 4 greater and we get a new maximum $\bar{e}_{54} = 0.2537 + \Delta w$. We have to make it lower and there are two ways: to decrease $\bar{w}_4$ or to increase $\bar{w}_5$. First way will undo last action, so we increase $\bar{w}_5$ and get $\bar{e}_{54} = 0.2537$, but also, we get $\bar{e}_{59} = -0.2537 - \Delta w$. It is bad again because we have this new $\max\limits_{i,j} |\bar{e}_{ij}| = 0.2537 + \Delta w$. We can decrease $\bar{w}_9$ by Δw and get $\bar{e}_{59} = -0.2537$. But $\bar{e}_{49}$ returns to its original value, from which we

Table 18.5 Alternatives' weight vector and corresponding upper bound proximity matrix

	ODC	ACC	QQRC	ADIC	APIC	LRC	ITC	CTC	ISC	CLC	OSC
i	0	1	2	3	4	5	6	7	8	9	10
$\bar{W}$	1.3663	0.0031	1.1086	0.9192	0.6176	1.1584	1.2354	-0.6922	-1.0372	-1.1414	-1.3971
$\bar{E}$	ODC	ACC	QQRC	ADIC	APIC	LRC	ITC	CTC	ISC	CLC	OSC
ODC	0	0.0070	0	0	0.0458	0	0	–0.0458	0	0	0
ACC	0	0	0	0	0	0	0	0.0992	0.0289	0.1801	0.0713
QQRC	0	0.1370	0	0	0	0	0	0.1066	–0.1370	0	0
ADIC	0	0.0811	0	0	0	0	0	0.0892	0.1191	0.0443	–0.1191
APIC	0	0.1801	0	0	0	0	0	0.2439	0.1370	0.2537	0
LRC	0	–0.0566	0	0	0.2537	0	0	–0.0202	–0.0612	–0.2537	0
ITC	0	0.0102	0	0	0.0753	0	0	0.0517	-0.0753	0	0
CTC	0	0	0	0	0	0	0	0	0	0.2439	0.1191
ISC	0	0	0	0	0	0	0	0	0	0	0
CLC	0	0	0	0	0	0	0	0	0	0	0
OSC	0	0	0	0	0	0	0	0	0	0	0

Table 18.6 Contender edges for removing from preferences graph

Coordinates, (i, j)	$\bar{e}_{ij}$	$\bar{r}_{ij}$	$\lvert\bar{w}_i - \bar{w}_j\rvert$
(4, 9)	0.2537	−0.0085	1.759
(4, 5)	0.2537	0.5408	0.5408
(5, 9)	−0.2537	0.3205	2.2998

started. Thus, we observe a cycle characterizing a violation of the consistency of the pairwise comparisons matrix. The only way to solve this cycle is to remove one of the edges $\bar{e}_{49}$, $\bar{e}_{54}$ or $\bar{e}_{59}$. Edges can be characterized by corresponding values such as values from upper bound proximity matrix, lower bound proximity matrix, and alternatives' weights or their combinations.

The choice of a candidate for deletion can be made in different ways. However, for each candidate we estimate $\lvert\bar{w}_i - \bar{w}_j\rvert$—this is a value characterizing the dependence on other edges. If this value is low, it means that in such graph configuration we decrease row i almost as many times as we increase column j to get balance—we call it the greatest inconsistency. By this principle, we remove edges until there are only edges characterizing the consistent solution adjoining the upper bound of the estimates. Note that if removing a particular edge violates the connectivity of the graph, the next suitable removes.

As a result, we obtain a solution, where $\max\limits_{i,j}\lvert\bar{e}_{ij}\rvert < \varepsilon$. Alternatives' weights vector is the upper solution. Weights vector and lower bound proximity matrix corresponding to it are presented in Table 18.7. The remaining edges of the graph are colored.

Lower solution making. Using the upper solution, we make a lower solution. For that, we "move" it to lower bound. First, we fix weights values and supplement matrix

Table 18.7 Alternatives' weights and lower bound proximity matrix corresponding to the upper solution

	ODC	ACC	QQRC	ADIC	APIC	LRC	ITC	CTC	ISC	CLC	OSC
$\overline{W^T}$	1.2089	0.3282	0.9871	1.0539	1.0794	1.1127	1.1756	-0.8037	-1.0217	-0.9333	-1.1434
$\overline{R^T}$	ODC	ACC	QQRC	ADIC	APIC	LRC	ITC	CTC	ISC	CLC	OSC
ODC	0	0	0	0	0	0	0	0.1438	0	0	0
ACC	0	0	0	0	0	0	0	0	0	0	0.4024
QQRC	0	0	0	0	0	0	0	0	0.1399	0	0
ADIC	0	0	0	0	0	0	0	0	0.4479	0	0.1642
APIC	0	0	0	0	0	0	0	0	0	0.2452	0
LRC	0	0	0	0	0	0	0	0	0.3497	0.0668	0
ITC	0	0	0	0	0	0	0	0.1560	0.2027	0	0
CTC	0	0	0	0	0	0	0	0	0	0	0
ISC	0	0	0	0	0	0	0	0	0	0	0
CLC	0	0	0	0	0	0	0	0	0	0	0
OSC	0	0	0	0	0	0	0	0	0	0	0

$\bar{R}^T$ with values, where $\bar{R} \geq 0$. This way we get a preferences graph G^B. Edges of this graph are colored in Table 18.8. Next, we find shift amount h. In process, we can delete some edges from G^B to get maximal h, but we should not break graph connectivity. This idea can be expressed by Eq. 18.7.

Table 18.8 Elementary moving weight vector and elementary moving matrix

	ODC	ACC	QQRC	ADIC	APIC	LRC	ITC	CTC	ISC	CLC	OSC
U	–2	–2	–2	–2	–3	–4	–2	–1	0	0	0
Y	ODC	ACC	QQRC	ADIC	APIC	LRC	ITC	CTC	ISC	CLC	OSC
ODC	0	0	0	0	0	0	0	1	0	0	0
ACC	0	0	0	0	0	0	0	1	2	2	2
QQRC	0	0	0	0	0	0	0	1	2	0	0
ADIC	0	0	0	0	0	0	0	1	2	2	2
APIC	0	1	0	0	0	0	0	2	3	3	0
LRC	0	0	0	0	1	0	0	3	4	4	0
ITC	0	0	0	0	0	0	0	1	2	0	0
CTC	0	0	0	0	0	0	0	0	0	1	1
ISC	0	0	0	0	0	0	0	0	0	0	0
CLC	0	0	0	0	0	0	0	0	0	0	0
OSC	0	0	0	0	0	0	0	0	0	0	0

$$\begin{cases} \bar{r}_{ij}^{B} = \mathrm{g}_{ij}^{\mathrm{B}} \cdot \left(\bar{r}_{ij}^{T*} - h \times y_{ij}\right) \\ \bar{w}_i^{B} = \bar{w}_i^{T} + h \times u_i \\ \min\limits_{i,j} \bar{r}_{ij}^{B} = 0 \\ h \to \max \end{cases} \tag{18.7}$$

Here, y_{ij} are elements of elementary moving matrix Y, u_i are elements of elementary moving weight vector U. These matrix and vector define from the structure of graph G^B. They reflect how alternatives' weights vector and relations matrix should be changed taking into account the transitivity of pairwise comparisons matrix. In Table 18.8, vector U and matrix Y corresponding to graph G^B is presented.

Thus, the maximum shift of the upper solution to the lower bound is obtained and equal $h = 0.14$. Alternative weight corresponding to the upper solution is calculated by Eq. 18.8.

$$\bar{w}_i^{B} = \bar{w}_i^{T} + h \times u_i \tag{18.8}$$

Middle solution construction. This solution is made from upper solution and the aforementioned shift value. First, we calculate shift for this solution as half of maximum shift $h_{\mathrm{mid}} = h/2$. This solution is equidistant from upper and lower solutions and it is more appropriate to use for obtaining numerical solution (as opposed to interval solution) provided by Eq. 18.9.

$$\bar{w}_i = \bar{w}_i^{T} + h_{\mathrm{mid}} \times u_i \tag{18.9}$$

Calculation normalized alternatives' weights in a linear scale. For obtaining final result, it is necessary to transition from a logarithmic scale to linear. It is easy to do using Eq. 18.10.

$$W = (w_i) = \left(\frac{e^{\bar{w}_i}}{\sum_{j=1}^{n} e^{\bar{w}_j}} \right) \tag{18.10}$$

For upper, lower, and middle alternatives weights normalized values are presented in Table 18.9. Note that for not normalized alternatives' weight vector in logarithmic scale according to Eqs. 18.7–18.9, the expression $\bar{w}_i^{B} \le \bar{w}_i \le \bar{w}_i^{T}$ is always true, but for normalized vector inequality of weights is usually not satisfied that results to alternatives ranking may change like in Table 18.9.

18.3.5 Results' Analysis

As a result, we got the weights of criteria for selection to MAH. There are two most able to influence the risk of airport modernization:

Table 18.9 Criteria weights for selection to MAH

No.	Criteria	Acronyms	W^T	W^B	W
1	Optimal distance from the center of Moscow	ODC	0.1593	0.1648	0.1624
2	Airport capacity	ACC	0.066	0.0683	0.0673
3	Quality and quantity of runways	QQRC	0.1276	0.1320	0.1301
4	Aerodrome infrastructure	ADIC	0.1364	0.1412	0.1390
5	Airport infrastructure	APIC	0.1399	0.1259	0.1330
6	Land resources	LRC	0.1446	0.1132	0.1282
7	Infrastructure of other transport	ITC	0.154	0.1594	0.1570
8	Cargo terminal quality	CTC	0.0213	0.0253	0.0233
9	International status	ISC	0.0171	0.0234	0.0201
10	Co-location	CLC	0.0187	0.0256	0.0219
11	Ownership form	OSC	0.0152	0.0207	0.0178

- Optimal distance from the center of Moscow.
- Infrastructure of other transport.

Evaluation of these parameters should be taken more carefully in further research. Also, the presence of the criterion of optimal distance from the center of Moscow in this list confirms the correctness of imposing restrictions on the estimated airports by those in a radius of 300 km.

The most unimportant criteria are identified too:

- Ownership form.
- International status.
- Co-location.
- Cargo terminal quality.

They may not be considered in further researches of the problem because of their insignificance, cargo terminal characteristics, presence of international status, co-location criterion, form of ownership and, to some extent, airport capacity.

18.4 Conclusions

The obtained criterion weights were used to calculate the integral risk of airport modernization in order to select MAH candidate airports. In addition to the application of criteria for the selection and evaluation of the contribution of the criteria, the results obtained are useful in themselves: the criteria that can most affect the risk of modernizing the airport, and those that cannot be considered in further research of the problem, were clearly distinguished.

In addition to the practical result, the applied method of incomplete pairwise comparisons with interval-specified estimates of alternative preferences deserves special attention. Its distinctive features are:

- Ability to operate with incomplete pairwise comparisons matrices without restoring missing estimates.
- Ability to work with both numerical and interval estimates of alternatives' preferences.

These advantages make it possible to apply the method of pairwise comparisons in complex cases like the following:

- When highly specialized experts cannot rate all alternatives, they rate alternatives of their expertise and for the rest, one give an interval or nothing at all.
- When it is required to compare a large number of alternatives and get the alternatives' weights from the resulting matrix of large dimension and incomplete matrices.

References

1. Borzova, A.S., Zheleznaya, I.P.: The development of regional airport infrastructure. Civ. Aviat. High Technol. **217**, 23–26 (2015)
2. Borzova, A.S., Zheleznaya, I.P.: Analysis of the state of the infrastructure of the airports of the Moscow aviation hub. Civ. Aviat. High Technol. **197**, 138–146 (2013)
3. Arakelyan, K.M.: The development model of the Moscow aviation hub. Civ. Aviat. **3**(7), 46–49 (2013)
4. Borodulina, S., Sokolov, V., Okuneva, A.: Passenger traffic forecasting logistics in air transport taking into account the influence of regional factors. Logistics **4**(101), 34–39 (2015)
5. Artobolevskiy, I.I., Russman, I.B., Sergeev, V.I., Statnikov, R.B.: On some methods of choosing an integral quality criterion in problems of optimal design of machines. J. Russ. Acad. Sci. USSR **2**, 3–10 (1978)
6. Saaty, T.L.: Decision Making for Leaders. The Analytic Hierarchy Process for Decisions in a Complex world. RWS Publications, Pittsburgh, Pennsylvania (2008)
7. Bochkov, A.V., Zhigirev, N.N., Ridley, A.N.: Method of recovery of priority vector for alternatives under uncertainty or incomplete expert assessment. Dependability **17**(3), 41–48 (2017)
8. Kuzmina, N.M., Ridley, A.N.: A method for estimating the impact of the fare rules conditions. Civ. Aviat. High Technol. **224**(2), 138–146 (2016)
9. Bochkov, A.V., Zhigirev, N.N.: Development of computation algorithm and ranking methods for decision-making under uncertainty. In: Ram, M., Davim, J. (eds.) Advanced Mathematical Techniques in Engineering Science. Series: Science, Technology and Management, pp. 121–154. CRC Press (2018)

Chapter 19
Adaptive Interpolation, TT-Decomposition and Sparse Grids for Modeling Dynamic Systems with Interval Parameters

Alexander Yu. Morozov and Dmitry L. Reviznikov

Abstract Problems with uncertainties arise in many practical fields and traditionally are formulated as dynamic systems with interval parameters. Often the complexity of existing methods is exponential in relation to the number of interval parameters. The adaptive interpolation algorithm and approaches directed to reducing the curse of dimensionality are considered. The main assumption on which these approaches are based is that not all interval parameters make a significant contribution to the solution of the problem. The use of tensor train decomposition and sparse grids allows us to take into account these features and expand the scope of the algorithm for the case of a large number of interval parameters. The effectiveness of the considered approaches is confirmed on several model problems.

19.1 Introduction

Problems with inaccurate data appear in many important areas of modern science. In particular, when solving various applied problems of the aerospace industry, problems of mechanics, and others, the situations often occur when some parameters are not exactly known, but there is information about the ranges, in which their values are located. Since most problems are formulated as a system of Ordinary Differential Equations (ODEs), it becomes necessary to solve the Cauchy problem with interval initial conditions or parameters [1].

A. Yu. Morozov · D. L. Reviznikov (✉)
Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, Moscow 125993, Russian Federation
e-mail: reviznikov@mai.ru

A. Yu. Morozov
e-mail: morozov@infway.ru

Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, 44, Vavilov ul., Moscow 119333, Russian Federation

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_19

Often, the complexity of existing methods is exponential in relation to the number of interval parameters. In this chapter, the adaptive interpolation algorithm [2–4] for modeling dynamic systems with interval parameters and approaches directed to reducing the curse of dimensionality are considered The main assumption, on which these approaches are based, is that not all interval parameters make a significant contribution to solve the problem. As a result, during the operation of the algorithm, data structures that have hidden dependencies and redundancy appear. Eliminating these structures can significantly reduce computational complexity.

The main idea of the adaptive interpolation algorithm is to build an adaptive hierarchical grid based on the kd-tree, in which each cell contains an interpolation grid, over the set formed by the interval initial conditions and parameters of the problem. For each time moment, an adaptive reconstruction of the partition is performed depending on the features of the solution. The result of the algorithm at each step is a piecewise polynomial function that interpolates the dependence of the solution on the parameter values with a given accuracy. Each vertex of a tree corresponds to a multidimensional array (or, according to the terminology in [5, 6], a tensor), for storage of which various effective representations can be used, under the assumption that the data have redundancy.

One of the effective representations of tensors is Tensor Train (TT) decomposition [5], which allows one to significantly reduce the amount of stored data in practice. This decomposition can be constructed using TT-cross algorithm [6] without calculating all the elements of the tensor. An important property is that all arithmetic (and not only) operations on tensors can be performed in this form.

The sparse grid method appeared in the 1960s [7] for solving multi-parameter problems in economics. It is used to interpolate the functions of many variables. Interpolation on sparse grids requires a significantly smaller number of nodes than conventional interpolation on a full grid. In this approach, instead of one dense grid, a linear combination of several sparse grids is used.

The chapter has the following organization. In Sect. 19.2, an interval statement of the Cauchy problem for a system of ODEs is presented. Section 19.3 is devoted to the adaptive interpolation algorithm for solving dynamic systems with interval parameters. In Sect. 19.4, we describe the problem of large dimensions and outline a way to a solution. In Sect. 19.5, the cross approximation of matrices is considered, which is the basis of tensor trains decomposition described in Sect. 19.6. This approach allows us to effectively deal with large dimensions. A modification of the adaptive interpolation algorithm using tensor trains is given in Sect. 19.7. Section 19.8 of the chapter is devoted to another approach to solving the problem of large dimensions—sparse grids. In Sect. 19.9, the main results are formulated.

19.2 Formulation of the Problem

We consider the Cauchy problem with interval initial conditions in the form of a system (Eq. 19.1).

$$\begin{cases} \dfrac{\mathrm{d}x_i(t)}{\mathrm{d}t} = f_i(x_1(t), x_2(t), \ldots, x_n(t)) \quad i = \overline{1, n} \\ x_i(t_0) \in \left[\underline{x_i^0},\ \overline{x_i^0}\right] \quad i = \overline{1, m} \\ x_i(t_0) = x_i^0 \quad i = \overline{m+1, n} \\ t \in [t_0,\ t_N] \end{cases} \tag{19.1}$$

If the ODE system is not autonomous or contains parameters, then dummy equations are added to the system so that it takes the form of Eq. 19.1. Function vector $\mathbf{f} = (f_1, f_2, \ldots, f_n)^{\mathrm{T}}$ satisfies all conditions ensuring the uniqueness and existence of a solution for all $\mathbf{x}(t_0) \in \left[\underline{\mathbf{x^0}},\ \overline{\mathbf{x^0}}\right]$.

The goal is to construct a piecewise polynomial vector function $\mathbf{P^k}\left(\mathbf{x^0}\right)$ for every time moment t_k, where $\mathbf{x^0} \in \left[\underline{\mathbf{x^0}},\ \overline{\mathbf{x^0}}\right]$, which interpolates the dependence of the solution on interval parameters with controlled accuracy. If a function $\mathbf{P^k}$ is found, an interval estimate of the solution (finding the left and right boundaries of the intervals) is reduced to a solution of $2n$ conditional optimization problems for an explicitly defined function.

19.3 Adaptive Interpolation Algorithm

Consider an adaptive interpolation algorithm with a regular grid at each vertex of the kd-tree. Assume that at the moment t_k there is a known solution $\mathbf{x^k}(\mathbf{x}^0)$. A grid G_0^k, which corresponds to the root vertex of the kd-tree and represents $(m+1)$-dimensional array, is constructed over the set formed by the interval initial conditions:

$$G_0^k[i_1, i_2, \ldots, i_m, j] = x_j^k\left(\underline{x_1^0} + \frac{\overline{x_1^0} - \underline{x_1^0}}{p} i_1,\ \underline{x_2^0} + \frac{\overline{x_2^0} - \underline{x_2^0}}{p} i_2, \ldots,\ \underline{x_m^0} + \frac{\overline{x_m^0} - \underline{x_m^0}}{p} i_m\right), \quad i_1, i_2, \ldots, i_m = \overline{0, p} \quad j = \overline{1, n},$$

where p is the degree of interpolation polynomial for each variable.

With function G_0^k, there is constructed G_0^{k+1}, which reduces to solve non-interval Cauchy problems for the corresponding elements provided by Eqs. 19.2.

$$\begin{cases} \dfrac{\mathrm{d}x_j(t)}{\mathrm{d}t} = f_j(x_1(t), x_2(t), \ldots, x_n(t)) \\ x_j(t_k) = G_0^k[i_1,\ i_2,\ \ldots, i_m, j] \\ G_0^{k+1}[i_1,\ i_2,\ \ldots, i_m, j] = x_j(t_{k+1}) \qquad j = \overline{1, n} \\ t \in \left[t_k,\ t_{k+1}\right] \end{cases} \tag{19.2}$$

System of Eqs. 19.2 can be considered as a function, the input of which is $G_0^k[i_1,\ i_2, \ldots, i_m]$ and the output is $G_0^{k+1}[i_1,\ i_2, \ldots, i_m]$.

With G_0^{k+1} an interpolation, polynomial $\mathbf{P^{k+1}}\left(\mathbf{x^0}\right)$ is constructed, for example, in the form of a Lagrange:

$$P_j^{k+1}\left(\mathbf{x^0}\right) = \sum_{i_1,i_2,\ldots,i_m=0}^{p} L\left(\mathbf{x^0}\right)[i_1, i_2, \ldots, i_m] \cdot G_0^{k+1}[i_1, i_2, \ldots, i_m, j], \quad j = \overline{1,n},$$

where $L\left(\mathbf{x^0}\right)$ is m-dimensional array that consists of the values of the basic Lagrange polynomials:

$$L\left(\mathbf{x^0}\right)[i_1, i_2, \ldots, i_m] = \prod_{j=0}^{p} \prod_{\substack{k=0 \\ i_k \neq j}}^{m} \frac{p\left(x_k^0 - \underline{x_k^0}\right) \Big/ \left(\overline{x_k^0} - \underline{x_k^0}\right) - j}{i_k - j}.$$

If the posterior error of interpolation

$$\text{error} = \max_{\mathbf{x^0} \in \left[\underline{\mathbf{x^0}},\ \overline{\mathbf{x^0}}\right]} \left\| \mathbf{x^{k+1}}(\mathbf{x^0}) - \mathbf{P^{k+1}}(\mathbf{x^0}) \right\| \tag{19.3}$$

is greater than some given value ε, than G_0^k is split into two grids G_1^k and G_2^k so that their estimate of the interpolation error is less than error. All the same actions are performed for them as for the grid G_0^k, and, if necessary, they are also broken.

As a result, at the time of t_{k+1} a kd-tree and the corresponding piecewise polynomial function that interpolates the solution with a given accuracy will be obtained. The process of constructing a kd-tree is illustrated in Fig. 19.1. There is no need to build a kd-tree from scratch at each step, instead, the tree obtained in the previous step is used, and depending on the estimate of the interpolation error, it is rebuilt. The process of crushing vertices always occurs at the previous step (dashed lines), because, when creating new vertices, the values associated with their nodes are interpolated, which must be performed at a time when the error is still valid. If the interpolation error becomes acceptable for the vertex and all its descendants, then the descendants are deleted, and the vertex itself becomes a leaf.

Assessment in the form of Eq. 19.3 in practice is not performed for all points from the region of uncertainty, but only for some points. When creating a vertex G_0^k, a test set of points is randomly created:

$$X_0^k = \left\{ \mathbf{x^k}(\hat{\mathbf{x}}^\mathbf{0}) \middle| \hat{\mathbf{x}}^\mathbf{0} = \text{rand}\left[\underline{\mathbf{x}^\mathbf{0}},\ \overline{\mathbf{x}^\mathbf{0}}\right] \right\}.$$

By analogy with the construction of G_0^{k+1}, X_0^{k+1} is constructed using X_0^k by solving non-interval Cauchy problems similar to Eq. 19.2. The posterior estimate of the interpolation error takes the following form:

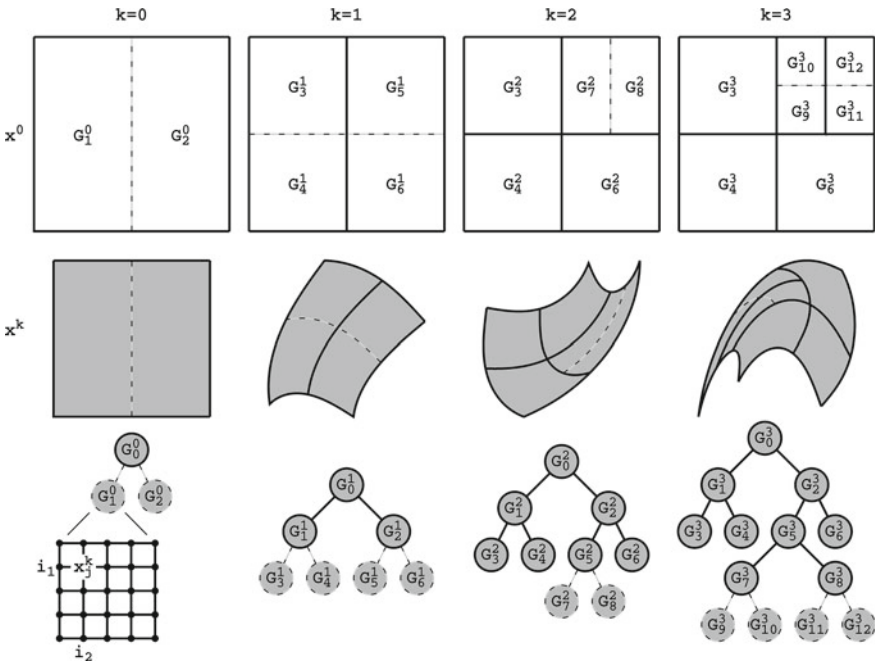


Fig. 19.1 Illustration of the working algorithm

$$\text{error} = \max_{(\hat{\mathbf{x}}^{\mathbf{0}},\, \mathbf{x}^{\mathbf{k+1}}) \in X_0^{k+1}} \left\| \mathbf{x}^{\mathbf{k+1}} - \mathbf{P}^{\mathbf{k+1}}(\hat{\mathbf{x}}^{\mathbf{0}}) \right\|.$$

The adaptive interpolation algorithm consists of three elements: transferring decisions to the next time layer, estimating the interpolation error for each vertex, and splitting the vertices. The considering approach is invariant in relation to specific implementations. It is not necessary, for example, to store explicitly multidimensional arrays at each vertex of the kd-tree, instead, it is enough to store only their TT-decomposition and implement all the necessary actions within TT-format. Also, it is not necessary to use the dense regular grids.

19.4 Large Dimensions

The described adaptive interpolation algorithm with all its advantages (universality, robustness, accuracy, and the possibility of parallelization) has one significant drawback: with an increase in the number of interval parameters, its complexity grows exponentially. Each vertex of the kd-tree contains a grid with the number of nodes $(p+1)^m$, where p is the degree of interpolation polynomial in each dimension and m is the number of interval initial conditions. Already with the parameters $p = 4$ and

$m = 20$ the number of nodes in the grid will be about one hundred trillion ($\approx 10^{14}$). In particular, a large number of interval parameters appears in such applied problem as modeling chemical transformations in the presence of uncertainties in the rate constants of reactions. If we consider the complete kinetic mechanisms in which the reactions are in the thousands, then we have to deal with the thousand-dimensional regions of parameter uncertainty, and the use of the adaptive interpolation algorithm, in this case, becomes impossible.

This situation can be improved by making an assumption that is fully consistent with reality. In practice, there is rarely a situation where absolutely all parameters individually or in combination have a significant impact on the solution. When constructing an interpolation polynomial, for example, for a function of two variables:

$$P(x, y) = a_{0,0} + a_{1,0}x + a_{0,1}y + a_{1,1}xy + \cdots = \sum_{i=0}^{p}\sum_{j=0}^{p} a_{i,j}x^i y^j.$$

This means that most of the members $a_{i,j}x^i y^j$ will not make a significant contribution to the result and, therefore, they can be ignored. Therefore, to construct an interpolation polynomial, it is sufficient to use not all $(p+1)^2$ nodes. There appear a few questions. Which terms need to be considered? How are the grid nodes selected? A priori, one cannot answer these questions unequivocally.

19.5 Cross Approximation

Let us consider the Lagrange interpolation polynomial on a regular grid for two variables. The calculation of the value at a certain point reduces to the elementwise multiplication of two matrices and the summation of all elements:

$$P(x, y) = L(x, y) \otimes F = \sum_{i=0}^{p}\sum_{j=0}^{p} l_{i,j}(x, y) f_{i,j},$$

where $l_{i,j}(x, y)$ are the basic Lagrange polynomials, $f_{i,j} = f(x_i, y_j)$ are values of the desired function in grid nodes x_i, y_j. The main goal is to reduce the number of calculations of function f.

The assumption made in the previous section essentially means that the rank of the matrix F may be less than $(p+1)$. Assume that the function f has the following structure:

$$f(x, y) = u_1(x)v_1(y) + u_2(x)v_2(y) + \cdots + u_r(x)v_r(y),$$

then the value matrix $F = \left\{f_{i,j}\right\}$ can be represented as the following decomposition:

$$F = \sum_{i=1}^{r} \begin{pmatrix} u_i(x_0) \\ u_i(x_1) \\ \dots \\ u_i(x_p) \end{pmatrix} \begin{pmatrix} v_i(y_0) & v_i(y_1) & \dots & v_i(y_p) \end{pmatrix} = UV.$$

Important fact is the following. If the matrix has a rank r, then if knowing rows r and columns r, we can completely restore the entire matrix. A number of questions arise. Which rows and columns to take? How to determine the rank of r? Of course, without any information about function F, it is impossible to answer these questions without calculating all the elements of the matrix; therefore, to some extent, all algorithms are heuristic. In practice, the transition from exact decomposition to approximation is performed with some accuracy ε:

$$\|F - UV\| < \varepsilon.$$

A number of works dedicated to methods for constructing such decomposition are known [8, 9]. The following methods such as skeletal decomposition, cross approximation, and low-rank approximation are found in the literature.

In its simplest form, the pseudo-code of the algorithm is presented in Fig. 19.2. The input is a matrix F of size $n \times m$, as well as, the required absolute accuracy of the approximation eps. The output is two matrices U and V of size $n \times r$ and $r \times m$ respectively, where r is the matrix rank, which is determined in the process of computing. The algorithm begins by selecting an arbitrary column (for example, with a number $m/2$). Next, the zeroing of a certain column or a certain row is alternately performed until the module of the maximum element becomes smaller than eps.

Consider the ODE system:

$$\begin{cases} x' = y, \quad y' = -\sin(x), \\ x(0) = x_0 \in [-1.0,\ 1.0], \\ y(0) = y_0 \in [0.0,\ 1.0]. \end{cases}$$

A regular grid is introduced over the set formed by the interval initial conditions: $x_0^i = -1 + 0.02i$, $y_0^j = 0.01j$, $0 \le i \le 100$, $0 \le j \le 100$. Let the matrix $\underset{101\times101}{F}$ consist of the values of the phase variable x at a time $t = 30$:

$$f_{i,j} = x_{i,j}(30) \ \mid \ x_0 = -1 + 0.02i,\ y_0 = 0.01j.$$

In Fig. 19.3, lines show those rows and columns that were needed in the process of constructing the decomposition. If the entire matrix consists of 10,201 elements, then it was necessary to calculate a total of 2101 elements (11 rows and 11 columns: $2101 = 2 \times 11 \times 101 - 11 \times 11$) that is a fifth of the entire matrix to construct an approximation with $eps = 10^{-5}$.

If we assume that all elements of the matrix are known, then the question of the existence of the decomposition and its finding is resolved: the SVD decomposition

```
r = 0, jr = m / 2
ii = {1, 2, ..., n}
jj = {1, 2, ..., m}
jj = jj / jr
while r < min(n, m):
     ir = -1
     for i = 1, ..., n:
            U[i][r] = F[i][jr];
            for k = 1, ..., r:
                   U[i][r] -= U[i][k] * V[k][jr]
            if (i in ii) and (ir < 0 or |U[i][r]| > |U[ir][r]|):
                   ir = i
     if ir < 0 or |U[ir][r]| < eps:
            break
     ii = ii / ir
     jr = -1
     for j = 1, ..., m:
            V[r][j] = F[ir][j]
            for k = 1, ..., r:
                   V[r][j] -= U[ir][k] * V[k][j]
            V[r][j] /= U[ir][r]
            if (j in jj) and (jr < 0 or |V[r][j]| > |V[r][jr]|):
                   jr = j
     if jr < 0 or |V[r][jr]| < eps:
            r += 1
            break
     jj = jj / jr
     r += 1
return U, V, r
```

Fig. 19.2 Pseudo-code of the algorithm

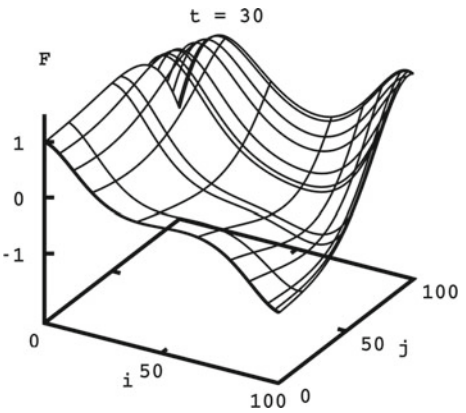


Fig. 19.3 Rows and columns, from which the entire matrix is restored

is uniquely constructed, where rows and columns corresponding to non-essential values of singular numbers are discarded.

19.6 Tensor Train

For matrices (in the case of two variables), the question is well studied and developed: there are many effective methods and algorithms for constructing decompositions. In the case of more than two measurements, it is necessary to work with multidimensional arrays, i.e., tensors. The general idea of efficient representing of these objects is based on the separation of variables. There are several approaches: canonical decomposition [10], Tucker decomposition [11], and TT-decomposition [5, 6] (tensor train). There are no reliable algorithms for the canonical decomposition, and the Tucker decomposition is difficult to apply for a large number of dimensions. TT-decomposition has appeared relatively recently, and its distinctive feature is the fact that it is not a subject to the curse of dimensionality.

Let $A \in \mathbb{R}^{p_1 \times p_2 \times \cdots \times p_n}$ be the given n-dimensional tensor. Its TT-decomposition is written as follows:

$$\hat{A}(i_1, i_2, \ldots, i_n) = G_1(i_1) G_2(i_2) \ldots G_n(i_n),$$

where

$$G_k \in \mathbb{R}^{p_k \times r_{k-1} \times r_k}, \; i_k = \overline{1, p_k}, \; k = \overline{1, n}, \quad r_0 = r_n = 1.$$

This is the product of $n - 2$ three-dimensional and two two-dimensional tensors (Fig. 19.4). To calculate the value of a particular element, the corresponding matrices are multiplied.

The construction of TT-decomposition reduces to the usual matrix decompositions. We perform the transition from n dimensions to 2 using index grouping. SVD-decomposition is calculated for the matrix. Rows and columns corresponding to non-essential singular numbers are discarded. The resulting matrices turn back into tensors of lower dimension. The algorithm is applied recursively for them. The idea of TT-cross algorithm that allows one to build TT-decomposition without calculating all

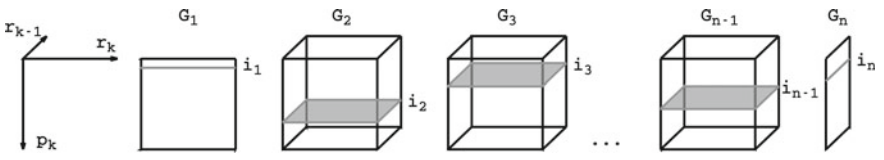


Fig. 19.4 Illustration of TT-decomposition

elements of the tensor is in replacing SVD-decomposition with decomposition with a lower computational cost which does not require full knowledge of all elements of the tensor.

In addition to not being subject to the curse of dimensionality, an important property of TT-decomposition is that most arithmetic operations, and not only arithmetic, can be performed on tensors, being in this format. This is, for example, finding the sum of all elements, determination of maximum/minimum, element-wise addition/subtraction/multiplication/division of two tensors, and so on. There are several implementations of TT-decomposition. We designed the main library `ttpy` using Python programming language. This library has all the necessary operations and methods for working with tensors in TT-format.

With the benefit of such tensors' representation, it became possible to work on ordinary computers with objects containing more elements than atoms in the Solar system. Of course, the important point is that the source data have tremendous redundancy, which is eliminated by this method.

19.7 Adaptive Interpolation Algorithm and TT-Decomposition

A multidimensional array (tensor) is stored at each vertex of the tree. The number of elements in the tensor depends on the number of interval parameters $(p+1)^m$. The main idea of practical improvement of this situation is to find TT-decomposition that can be constructed using the TT-cross algorithm without calculating all the elements of the tensor.

The adaptive interpolation algorithm conditionally consists of three actions: transferring all the solutions contained in the vertices of the kd-tree to the next time layer, interpolating along the grid, and splitting the vertex into two. The first action within each vertex can be considered as constructing a tensor consisting of the values of some function. This action can be effectively performed using TT-cross algorithm. The interpolation operation is reduced to elementwise multiplication of two tensors, one of which is composed of the values of the basic Lagrange polynomials, and the second is composed of the values of the interpolated function, followed by the summation of all elements. If we assume that the split of vertices is always performed by a hyperplane perpendicular to one of the coordinate axes, then one-dimensional interpolation is used to construct new vertices. In general, all the actions that make up the algorithm can be represented as compositions of several operations available in TT-format.

In the original version of the algorithm, the order p and, accordingly, the size of the interpolation grid was determined from specific considerations regarding stability and computational complexity. Here, in some cases, it is advisable to create a grid, where there will be more nodes than it is required for a given order p. Splitting a

vertex into two can be more expensive from a computational point of view than increasing the number of nodes within a single grid.

As an example, we consider the model problem: the motion of bodies with uncertainties in the initial velocities under the influence of gravitational forces. The ODE system in dimensionless variables has the following form:

$$\begin{cases} \left(v_i^x\right)' = \sum\limits_{j=1,\ j\neq i}^{7} m_j \dfrac{x_j - x_i}{r_{i,j}^3}, \quad \left(v_i^y\right)' = \sum\limits_{j=1,\ j\neq i}^{7} m_j \dfrac{y_j - y_i}{r_{i,j}^3}, \quad \left(v_i^z\right)' \\ \quad = \sum\limits_{j=1,\ j\neq i}^{7} m_j \dfrac{z_j - z_i}{r_{i,j}^3}, \\ x_i' = v_i^x, \ y_i' = v_i^y, \ z_i' = v_i^z, \ i = \overline{1,7}, \ t \in [0.0,\ 0.02], \\ x_1(0) = y_1(0) = z_1(0) = v_1^x(0) = v_1^y(0) = v_1^z(0) = 0, \\ x_{2,3}(0) = \pm 1, \ y_{2,3}(0) = z_{2,3}(0) = 0, \ v_{2,3}(0) = \left(0 \pm v\ 0\right)^T + \Delta v_{2,3}^T, \\ y_{4,5}(0) = \pm 1, \ x_{4,5}(0) = z_{4,5}(0) = 0, \ v_{4,5}(0) = \left(0\ 0 \pm v\right)^T + \Delta v_{4,5}^T, \\ z_{6,7}(0) = \pm 1, \ x_{6,7}(0) = y_{6,7}(0) = 0, \ v_{6,7}(0) = \left(\pm v\ 0\ 0\right)^T + \Delta v_{6,7}^T, \end{cases} \tag{19.4}$$

where $r_{i,j} = \sqrt{\left(x_j - x_i\right)^2 + \left(y_j - y_i\right)^2 + \left(z_j - z_i\right)^2}$ is the distance between two bodies, $v = 316.23$ is the initial velocity of bodies, $m_1 = 10^5$, $m_{\overline{2,7}} = 10^{-5}$ is the body mass, $\Delta v_{\overline{2,7}} = ([-2,\ 2],\ [-2,\ 2],\ [-2,\ 2])$ is the interval uncertainties in the velocities of bodies.

The solution to the problem is presented in Fig. 19.5. Rectangular parallelepipeds illustrate areas of uncertainty in space for each body at different time points. This system is indicative because the uncertainty in the speed of a particular body mainly affects only on the position and speed of that body and weakly affects on other bodies. The parameter $p = 4$, the number of elements in the tensor at each vertex is $5^{18} \cdot 42 \approx 10^{15}$. The number 42 corresponds to the number of phase variables.

Due to the redundancy in the data, only a small part of the elements of the initial tensor is required to construct TT-decomposition. In general, it is very effective to use TT-cross algorithm to reduce computational costs and apply the adaptive interpolation algorithm for problems with a large number of interval parameters.

19.8 Sparse Grids

One of the important approaches to reduce the curse of dimensionality is sparse Smolyak grids [7, 12–15]. They appeared in the 1960s solving the multi-parameter problems in economics. Interpolation uses a piecewise linear hierarchical basis (Fig. 19.6a) based on the hat function mentioned below.

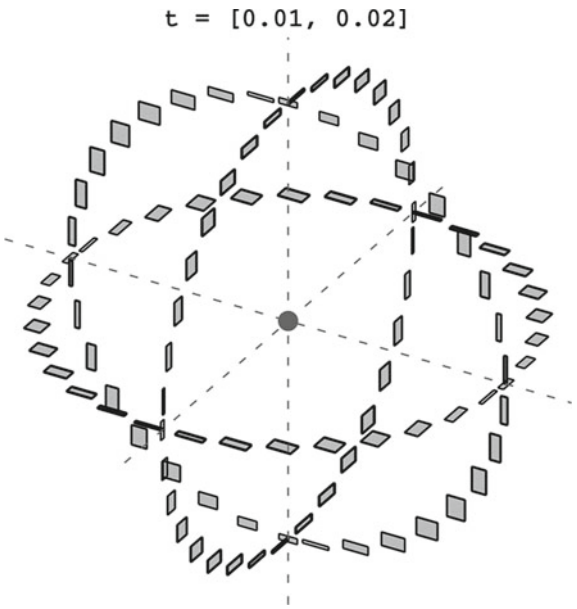


Fig. 19.5 Uncertainties in the position of bodies at different points in time

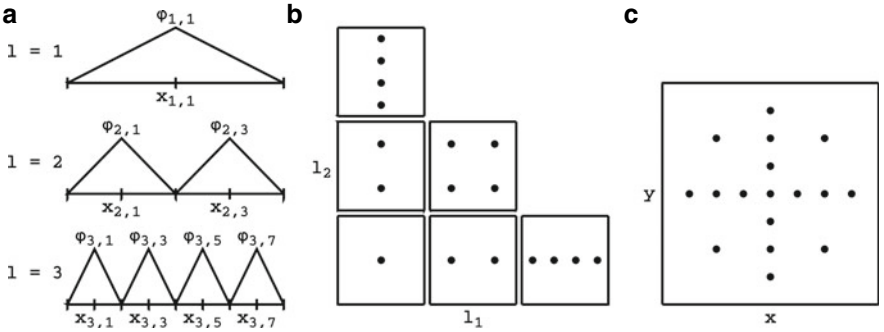


Fig. 19.6 Sparse grids. Hierarchical basis: **a** one-dimensional basis, **b** construction of a two-dimensional basis, **c** two-dimensional sparse grid

$$\varphi(x) = \begin{cases} 1 - |x|, & x \in [-1,\ 1] \\ 0, & \text{otherwise} \end{cases}$$

Consider a set of grids G_l on a unit interval $[0,\ 1]$, where l is the level that defines the width of the grid as $h_l = 2^{-l}$. The grid points $x_{l,\,i}$ are specified as:

$$x_{l,\,i} = i \cdot h_l,\ 0 \le i \le 2^l.$$

Families of basis functions $\varphi_{l,\,i}(x)$ are generated from the obtained sets of points using the stretching and shift of the hat function $\varphi(x)$:

$$\varphi_{l,\,i}(x) = \varphi\big((x - i \cdot h_l)\big/ h_l\big).$$

For each value l, the functions $\varphi_{l,\,i}(x)$ form a nodal basis (or Lagrange basis). It is obvious that

$$\operatorname{span}\big\{\varphi_{l,\,i} :\ 1 \le i \le 2^l - 1\big\} = \bigoplus_{k \le l} \operatorname{span}\big\{\varphi_{k,\,i} :\ 1 \le i \le 2^k - 1,\ i \text{ odd}\big\}.$$

In the case of a sparse grid (Fig. 19.6b, c), the d-dimensional basis of the level n is defined as

$$\varphi_{l_1 l_2 \ldots l_d, i_1 i_2 \ldots i_d}(x_1, x_2, \ldots, x_d) = \prod_{j=1}^{d} \varphi_{l_j,\,i_j}(x_j),$$

$$\sum_{j=1}^{d} l_j \le n + d - 1, \quad 1 \le i_j \le 2^{l_j} - 1, \quad i_j \text{ odd}.$$

Compared to the full grid ($\max\big(l_j\big) \le n$), the sparse grid has a significantly smaller number of nodes, but at the same time, asymptotically, the interpolation error rises n^{d-1} times.

There is an adaptive version of these grids, where a binary tree is used for structuring. In the classical version, if the investigated function has a non-zero value at the boundary of the region, then all faces of smaller dimensions are considered. The adaptive grid is built for each face. It is easy to calculate that for an n-dimensional region the number of faces of smaller dimension will be $3^n - 1$. Given the duplication of nodes in different grids, in the best case, the number of nodes will be equaled 3^n, which indicates the exponential complexity of this approach. An important property of sparse grids is the flexibility of adaptation: for a small increase in accuracy in each particular case, there is no need to immediately double the number of nodes, as would be necessary for the adaptive interpolation algorithm.

Consider examples. Figure 19.7 shows several functions $\mathbb{R}^2 \to \mathbb{R}$ and the resulting adaptive grid, and Fig. 19.8 shows the grids for functions $\mathbb{R}^3 \to \mathbb{R}$. These figures show that this approach defines the combinations of parameters that play a significant role, while the construction of grid does not occur on the entire set, but only on the subsets corresponding to these combinations.

The calculation of the approximate value of the function at a given point is reduced to the summation of the basic functions with certain weight coefficients, which are determined in accordance with the interpolated function. The values of these weights can be considered as the adaptation criterion, according to which the grid is compressed.

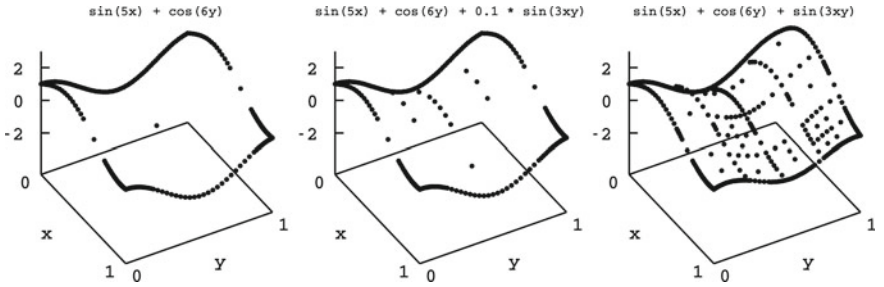


Fig. 19.7 Examples of interpolation of functions of two variables

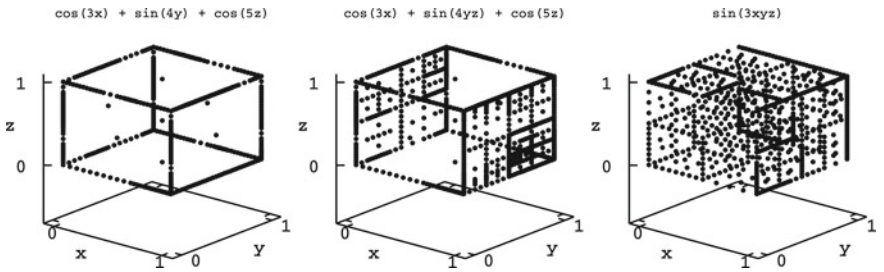


Fig. 19.8 Examples of grids for functions of three variables

One step of modeling a dynamic system with interval parameters can be written as follows:

1. Transfer of all solutions of the non-interval ODE system corresponding to the nodal points of the sparse grid to the next time layer.
2. Recalculation of weights.
3. Grid rebuilding.

Consider the ODE system describing Lotka–Volterra model with three interval initial conditions and seven interval parameters:

$$\begin{cases} x' = x(\delta_1 - y - \varepsilon x), \\ y' = -\gamma_1 y(\delta_2 - x + z) - \varphi y^2, \\ z' = -\gamma_2 z(\alpha - y), \end{cases} \quad \begin{array}{l} x(0),\ y(0),\ z(0),\ \delta_1,\ \delta_2,\ \gamma_1,\ \gamma_2 \in [1.0,\ 1.01], \\ \varepsilon,\ \varphi \in [-0.0005,\ 0.0005], \\ \alpha \in [0.9,\ 0.91]. \end{array}$$

Figure 19.9 shows the dependence of interval estimates of solutions in time.

The number of nodes at the end time equals 305,481 and posterior global error $\approx 10^{-2}$. For comparison, using TT-decomposition with parameter $p = 4$ led to the integration of 1,528,325 non-interval ODEs at the last step. Thus, for this problem, sparse grid approach appears to be more effective. This is partly due to the fact that in this case, all the parameters have a different effect on the solution, and the use of a dense grid is not the optimal way. Also, note that the number of parameters in the

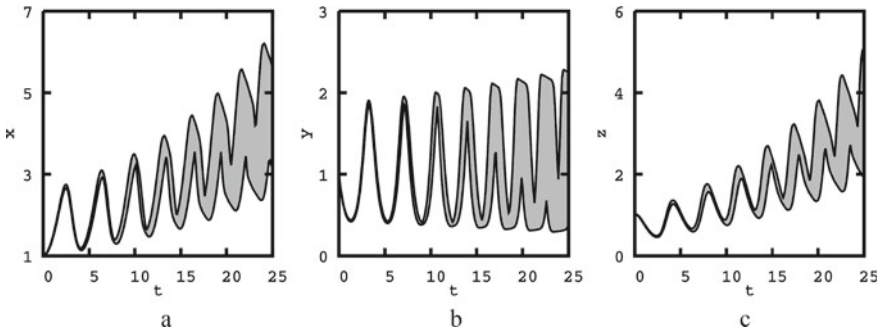


Fig. 19.9 The dependence of the upper and lower estimates of the solution on time: **a** internal estimate $x(t)$, **b** internal estimate $y(t)$, **c** internal estimate $z(t)$

considered problem (10 parameters) is a rather modest number. Increasing this value gives an advantage to the adaptive interpolation algorithm with TT-decomposition.

19.9 Conclusions

The approaches described in the chapter are aimed at reducing exponential complexity in solving multidimensional problems. All of them are based on the assumption that the desired function has a certain form. Some parameters may strongly affect the result, may have a weak effect, or may not affect at all. Automatically taking into account these features, it is possible to effectively reduce the complexity of the task.

The adaptive interpolation algorithm for modeling dynamic systems with interval parameters has been described. The algorithm has exponential complexity on the number of interval parameters, which limits its scope of usage. The influence of interval uncertainties on a solution can often be degenerate. TT-decomposition and sparse grids allow one to take this degeneracy into account and expand the scope of the algorithm to the case of a large number of interval parameters. The effectiveness of the considered approaches is confirmed on several model problems.

Acknowledgements The work was performed with the support of the Ministry of Science and Higher Education of the Russian Federation grant No. 2020-1902-01-016.

References

1. Moore, R.E.: Interval Analysis. Prentice Hall, Englewood Cliffs (1966)

2. Morozov, AYu., Reviznikov, D.L.: Adaptive interpolation algorithm based on a kd-tree for numerical integration of systems of ordinary differential equations with interval initial conditions. Diff. Eqn. **54**(7), 945–956 (2018)
3. Morozov, AYu., Reviznikov, D.L., Gidaspov, VYu.: Adaptive interpolation algorithm based on a kd-tree for the problems of chemical kinetics with interval parameters. Math. Models Comput. Simul. **11**(4), 622–633 (2019)
4. Morozov, AYu., Reviznikov, D.L.: Modelling of dynamic systems with interval parameters on graphic processors. Programmnaya Ingeneria **10**(2), 69–76 (2019)
5. Oseledets, I.V.: Tensor-train decomposition. SIAM J. Sci. Comput. **33**(5), 2295–2317 (2011)
6. Oseledets, I., Tyrtyshnikov, E.: TT-cross approximation for multidimensional arrays. Linear Algebra Appl. **432**(1), 70–88 (2010)
7. Smolyak, S.A.: Quadrature and interpolation formulas on tensor products of certain classes of functions]. Dokl. AN SSSR **148**(5), 1042–1045(in Russian) (1963)
8. Goreinov, S.A., Tyrtyshnikov, E.E., Zamarashkin. N.L.: A theory of pseudoskeleton approximations. Linear Algebra Appl. **261**(1–3), 1–21 (1997)
9. Tyrtyshnikov, E.: Incomplete cross approximation in the mosaic-skeleton method. Computing **64**(4), 367–380 (2000)
10. Hitchcock, F.L.: The expression of a tensor or a polyadic as a sum of products. J. Math. Phys. **6**(1), 164–189 (1927)
11. Tucker, L.R.: Some mathematical notes on three-mode factor analysis. Psychometrika **31**, 279–311 (1966)
12. Gerstner, T., Griebel, M.: Sparse grids. In: Cont, R. (ed.) Encyclopedia of Quantitative Finance, Wiley (2008)
13. Garcke, J.: Sparse grids in a nutshell. In: Garcke, J., Griebel, M. (eds.) Sparse Grids and Applications. LNCSE, vol. 88, pp. 57–80. Springer, Berlin, Heidelberg (2013)
14. Brumm, J., Scheidegger, S.: Using adaptive sparse grids to solve high-dimensional dynamic models. Econometrica **85**(5), 1575–1612 (2017)
15. Bungatrz, H.-J., Griebel, M.: Sparse grids. Acta Numerica **13**(1), 147–269 (2004)

Chapter 20
Using Spectral Form of Mathematical Description to Represent Iterated Stratonovich Stochastic Integrals

Konstantin A. Rybakov

Abstract In this chapter, it is suggested to apply the spectral form of mathematical description for the representation of the iterated Stratonovich stochastic integrals of an arbitrary multiplicity. Some invariant relations for expansion coefficients and the iterated Stratonovich stochastic integrals are obtained. An algorithm for modeling the iterated Stratonovich stochastic integrals is discussed.

20.1 Introduction

Iterated stochastic integrals play a fundamental role in the constructing high-order numerical methods for stochastic differential equations. These methods are based on the Taylor–Itô expansion and the Taylor–Stratonovich expansion for random processes. The first numerical method using iterated stochastic integrals of multiplicity 2 and orthogonal expansions into the trigonometric series was Milstein method [1–3]. Iterated stochastic integrals of multiplicity 3 and orthogonal expansions into the trigonometric series have also been used in [4] by Kloeden and Platen. In Kuznetsov method, various complete orthonormal systems may be applied for the representation of iterated stochastic integrals of an arbitrary multiplicity [5]. Orthogonal expansions into the trigonometric series and the Haar series for Milstein method have been investigated in [6]. The numerical simulation of iterated stochastic integrals is discussed in [2, 3, 7–9].

The paper [10] deals with orthogonal expansions for random processes with respect to Milstein method by using the spectral form of mathematical description. In this chapter, the application of the spectral form of mathematical description for the representation of the iterated Stratonovich stochastic integrals of an arbitrary

K. A. Rybakov (✉)
Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, Moscow 125993, Russian Federation
e-mail: rkoffice@mail.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_20

multiplicity is considered. The special case of Walsh series has been discussed in [11], and here it is suggested to apply orthogonal expansions with respect to arbitrary complete orthonormal systems.

Obtained results may be used in the constructing high-order numerical methods based on the Taylor–Stratonovich expansion for random processes [2–4, 12–15], and also in numerical methods based on Taylor–Itô expansion due to the known relationship between the iterated Itô and Stratonovich stochastic integrals [13, 15]. High-order numerical methods may be applied for the modeling stochastic dynamical systems [3, 4, 12, 16], the solving optimal and suboptimal filtering problems [17–19], and the optimizing dynamical systems of the joint estimation and control [20–23]. Iterated stochastic integrals can be used in the constructing high-order numerical methods for non-commutative semilinear stochastic partial differential equations [24].

The goal of this research is to obtain the representation of the iterated Stratonovich stochastic integrals using the spectral form of mathematical description of signals and control systems [25–28] and to construct the spectral method and corresponding algorithm for modeling the iterated Stratonovich stochastic integrals.

The rest of this chapter is structured as follows. Section 20.2 provides definitions of the iterated Itô and Stratonovich stochastic integrals. Elements of the spectral form of mathematical description are described in Sect. 20.3. The main result of the chapter, i.e., the representation of the iterated Stratonovich stochastic integrals using the spectral form of mathematical description, is given in Sect. 20.4. Further, some invariant relations based on results of Sect. 20.4 for expansion coefficients and the iterated Stratonovich stochastic integrals are obtained in Sect. 20.5. Section 20.6 gives the tensor representation for the expansion coefficients. Finally, Sect. 20.7 presents the conclusions for this chapter.

20.2 Iterated Itô and Stratonovich Stochastic Integrals

Let $(\Omega, \mathcal{F}, \mathrm{P})$ be a probability space, where Ω is the sample space, $\mathcal{F}$ is a σ-algebra of subsets of Ω, and P is a probability measure, and let $\mathcal{F}_t$ be a non-decreasing family of σ-subalgebras of $\mathcal{F}$, $t \geqslant 0$.

The iterated Stratonovich stochastic integrals of multiplicity $k \geqslant 2$ are defined as follows:

$$I_h^{*(j_1 j_2 \ldots j_k)} = \int\limits_0^h \ldots \int\limits_0^{\tau_3} \int\limits_0^{\tau_2} \mathrm{d}w_{j_1}(\tau_1) \circ \mathrm{d}w_{j_2}(\tau_2) \circ \ldots \circ \mathrm{d}w_{j_k}(\tau_k), \quad j_1, j_2, \ldots, j_k = 1, 2, \ldots, s, \tag{20.1}$$

where $h > 0$ and $w_1(t), w_2(t), \ldots, w_s(t)$ are $\mathcal{F}_t$-adapted independent standard Wiener processes. The integral of multiplicity $k = 1$ is a centered Gaussian random variable

$$I_h^{*(j_1)} = \int_0^h \mathrm{d}w_{j_1}(\tau) = w_{j_1}(h), \quad j_1 = 1, 2, \ldots, s,$$

with the second-order moment h. In this context, h is an integration step in numerical methods for stochastic differential equations [2–4, 12, 13, 15].

Note that for the case of pairwise distinct values $j_1, j_2, \ldots, j_k$ the iterated Stratonovich stochastic integral defined by Eq. 20.1 coincides with the corresponding iterated Itô stochastic integral

$$I_h^{(j_1 j_2 \ldots j_k)} = \int_0^h \ldots \int_0^{\tau_3} \int_0^{\tau_2} \mathrm{d}w_{j_1}(\tau_1)\mathrm{d}w_{j_2}(\tau_2) \ldots \mathrm{d}w_{j_k}(\tau_k),$$

and in the general case the relationship between the Itô and iterated Stratonovich stochastic integrals is described in [13, 15].

In [13, 15], it is proposed a general approach to the representation and modeling the iterated Itô and Stratonovich stochastic integrals. This approach is based on finding the expansion coefficients of the function

$$K(t_1, t_2, \ldots, t_k) = \begin{cases} 1 \text{ for } t_1 < t_2 < \ldots < t_k \\ 0 \qquad \text{otherwise} \end{cases} \tag{20.2}$$

with respect to the orthonormal basis $\{q(i_1, t_1)q(i_2, t_2) \ldots q(i_k, t_k)\}_{i_1, i_2, \ldots, i_k = 0}^{\infty}$ in $L_2([0, h]^k)$, where $\{q(i, t)\}_{i=0}^{\infty}$ is the orthonormal basis in $L_2([0, h])$. In [13, 15, 29], the Legendre polynomials and trigonometric functions are considered in detail for the representation and modeling iterated stochastic integrals. Moreover, the general case of the function $K(t_1, t_2, \ldots, t_k)$ and corresponding iterated stochastic integrals are discussed. Representations for iterated stochastic integrals of multiplicity 1 and 2 have also been obtained for the Walsh and Haar functions [6, 15].

The spectral form of mathematical description has been used in [10] for iterated stochastic integrals of multiplicity 1 and 2 with respect to the Legendre polynomials and trigonometric functions as well as the Walsh and Haar functions. The case $k = 1$ is trivial, and we have $K(t_1, t_2) = 1(t_2 - t_1)$ for $k = 2$, where $1(t)$ is the unit step function that defines the impulse response function of the integrating element. Therefore, the iterated stochastic integral of multiplicity 2 may be represented using the spectral characteristic of the integration operator (two-dimensional nonstationary transfer function of the integrating element [25]). This spectral characteristic and also the spectral characteristic of the multiplier will be applied below for the case $k > 2$.

The following representation of the iterated Stratonovich stochastic integrals by the iterated series holds

$$I_h^{*(j_1 j_2 \ldots j_k)} = \sum_{i_k=0}^{\infty} \ldots \sum_{i_2=0}^{\infty} \sum_{i_1=0}^{\infty} C_{i_k \ldots i_2 i_1} \zeta_{i_1}^{(j_1)} \zeta_{i_2}^{(j_2)} \ldots \zeta_{i_k}^{(j_k)}, \tag{20.3}$$

where $C_{i_k \ldots i_2 i_1}$ are expansion coefficients of the function $K(t_1, t_2, \ldots, t_k)$, i.e.,

$$\begin{aligned} C_{i_k \ldots i_2 i_1} &= \int_0^h \ldots \int_0^h \int_0^h q(i_1, t_1) q(i_2, t_2) \ldots q(i_k, t_k) K(t_1, t_2, \ldots, t_k) \mathrm{d}t_1 \mathrm{d}t_2 \ldots \mathrm{d}t_k \\ &= \int_0^h q(i_k, \tau_k) \ldots \int_0^{\tau_3} q(i_2, \tau_2) \int_0^{\tau_2} q(i_1, \tau_1) \mathrm{d}\tau_1 \mathrm{d}\tau_2 \ldots \mathrm{d}\tau_k, \\ & i_1, i_2, \ldots, i_k = 0, 1, 2, \ldots, \end{aligned} \tag{20.4}$$

and $\zeta_i^{(j)}$ are independent random variables having a standard normal distribution, $j = 1, 2, \ldots, s$ and $i = 0, 1, 2, \ldots$.

A detailed proof of Eq. 20.3 for the case $k \leqslant 4$ and a discussion about the case $k \geqslant 5$ is given in [15]. Note that Eq. 20.3 and further relations for iterated stochastic integrals are understood with probability 1.

20.3 Elements of Spectral Form of Mathematical Description

Let $\{q(i,t)\}_{i=0}^{\infty}$ be a orthonormal basis in $L_2([0,h])$, $f(t)$ is an element from $L_2([0,h])$, i.e., $\int_0^h f^2(t)\mathrm{d}t < \infty$. Then the function $f(t)$ is determined by expansion coefficients represented as the infinite column matrix F with entries

$$F_i = \int_0^h q(i,t) f(t) \mathrm{d}t, \quad i = 0, 1, 2, \ldots, \tag{20.5}$$

i.e.,

$$f(t) = \sum_{i=0}^{\infty} F_i q(i,t), \quad t \in [0,h]. \tag{20.6}$$

To indicate the relationship between the function $f(t)$ and the infinite column matrix F, we will use notations $F = \mathbb{S}[f(t)]$ and $f(t) = \mathbb{S}^{-1}[F]$. According to [25], the infinite column matrix F is called the spectral characteristic of the function

$f(t)$ defined with respect to the orthonormal basis $\{q(i,t)\}_{i=0}^{\infty}$, $\mathbb{S}$ is the spectral transform and $\mathbb{S}^{-1}$ is the spectral inversion.

Further, we will use some properties of the spectral transform [25]. Let $x(t), y(t), z(t) \in L_2([0,h])$ and $X = \mathbb{S}[x(t)]$, $Y = \mathbb{S}[y(t)]$, $Z = \mathbb{S}[z(t)]$, then we have:

(1) If $y(t) = \dot{x}(t)$, $x(0) = 0$, then $Y = PX$, where P is the infinite matrix with entries

$$P_{i_1 i_2} = \int_0^h q(i_1,t)\dot{q}(i_2,t)\mathrm{d}t + q(i_1,0)q(i_2,0), \quad i_1, i_2 = 0, 1, 2, \ldots$$

(2) If $y(t) = \int_0^t x(\tau)d\tau$ $(t \leqslant h)$, then $Y = P^{-1}X$, where P^{-1} is the infinite matrix with entries

$$P^{-1}_{i_1 i_2} = \int_0^h q(i_1,t) \int_0^t q(i_2,\tau)\mathrm{d}\tau\mathrm{d}t, \quad i_1, i_2 = 0, 1, 2, \ldots$$

(3) If $z(t) = x(t)y(t)$, then $Z = (VX)Y$, where V is the three-dimensional infinite matrix with entries

$$V_{i_1 i_2 i_3} = \int_0^h q(i_1,t)q(i_2,t)q(i_3,t)\mathrm{d}t, \quad i_1, i_2, i_3 = 0, 1, 2, \ldots$$

The matrices P, P^{-1}, and V are called the spectral characteristic of the differentiation operator taking into account the initial condition, the spectral characteristic of the integration operator, and the spectral characteristic of the multiplier, respectively. It should be emphasized that $PP^{-1} = P^{-1}P = E$, where E is the infinite identity matrix, and V is the symmetric three-dimensional infinite matrix with respect to any pair of indices from the triple (i_1, i_2, i_3), i.e., any section of V is the infinite symmetric matrix.

It is important to note that the condition $x(t), y(t), z(t) \in L_2([0,h])$ is only sufficient to determine spectral characteristics of these functions and to fulfill above properties. In some cases, this condition can be weakened, e.g., we can indicate the relationship between the spectral characteristic $\mathbf{1}$ of the unit step function $1(t)$ and the infinite column matrix Δ, where entries of Δ are values of the orthonormal basis $\{q(i,t)\}_{i=0}^{\infty}$ at $t = 0$. The column matrix Δ is called the spectral characteristic of

Dirac delta function [25]. In fact, the spectral characteristic definition is formally extended to functions, for which the expansion coefficients can be calculated using Eq. 20.5. It defines the linear functional on the set of spectral characteristics of functions and $P^{-1}\Delta = \mathbf{1}$, $P\mathbf{1} = \Delta$ (if an additional condition for the pointwise convergence is fulfilled).

Moreover, it is possible to apply the spectral form of mathematical description not only for deterministic functions, but also for random processes. If $f(t)$ is a random process satisfying the condition $E\int\limits_0^h f^2(t)\mathrm{d}t < \infty$, where E is the expectation, then $f(t)$ is determined by random expansion coefficients, which are also represented as the infinite column matrix, and this matrix is called the spectral characteristic of the random process $f(t)$ defined with respect to the orthonormal basis $\{q(i,t)\}_{i=0}^{\infty}$. The spectral characteristic definition can be extended to a class of random processes, for which Eq. 20.5 is applicable. Thus, the spectral characteristic $\mathcal{V}$ of standard Gaussian white noise $v(t)$ is an infinite column matrix, whose entries are independent random variables having a standard normal distribution. It is related to the spectral characteristic $\mathcal{W}$ of the standard Wiener random process $w(t)$ as $P^{-1}\mathcal{V} = \mathcal{W}$, $P\mathcal{W} = \mathcal{V}$ [30, 31]. Note that the spectral characteristic $\mathcal{V}$ defines the random linear functional on the set of spectral characteristics of functions.

In addition, we should indicate one more property of the spectral transform:

(4) The spectral transform preserves the norm and the inner product, i.e.,

$$\int\limits_0^h x^2(t)\mathrm{d}t = X^{\mathrm{T}}X, \quad \int\limits_0^h x(t)y(t)\mathrm{d}t = X^{\mathrm{T}}Y.$$

As an example of using the spectral form of mathematical description, we will represent the spectral characteristic of the function

$$x_k(t) = \int\limits_0^t g_k(\tau_k)\ldots\int\limits_0^{\tau_3} g_2(\tau_2)\int\limits_0^{\tau_2} g_1(\tau_1)d\tau_1 d\tau_2\ldots d\tau_k \tag{20.7}$$

by spectral characteristics of functions $g_l(t)$, $l = 1, 2, \ldots, k$.

Consider a System of Ordinary Differential Equations (SODE):

$$\begin{aligned} &\dot{x}_1(t) = g_1(t),\ \dot{x}_2(t) = g_2(t)x_1(t),\ \ldots,\ \dot{x}_k(t) = g_k(t)x_{k-1}(t),\\ &x_1(0) = x_2(0) = \cdots = x_k(0) = 0. \end{aligned} \tag{20.8}$$

The spectral form of mathematical description was proposed for the dynamical systems analysis. In this context, SODE (Eq. 20.8) can be considered as a mathematical model of the dynamical system, for which functions $g_l(t)$ and $x_l(t)$ are input and output signals ($l = 1, 2, \ldots, k$), respectively.

The solution of SODE (Eq. 20.8) is formally obtained as a result of the sequential integration. Thus,

$$x_1(t) = \int_0^t g_1(\tau)\mathrm{d}\tau, x_2(t) = \int_0^t g_2(\tau)x_1(\tau)\mathrm{d}\tau, \ldots, x_k(t) = \int_0^t g_k(\tau)x_{k-1}(\tau)\mathrm{d}\tau, \tag{20.9}$$

consequently, we have Eq. 20.7.

Using properties 1–3 and introducing notations $G_l = \mathbb{S}[g_l(t)]$ and $X_l = \mathbb{S}[x_l(t)]$, we can write spectral analogs for Eqs. 20.8–20.9. Thus,

$$PX_1 = G_1, \quad PX_2 = (VG_2)X_1, \quad \ldots, \quad PX_k = (VG_k)X_{k-1}, \tag{20.10}$$

and

$$\begin{aligned} X_1 &= P^{-1}G_1, \\ X_2 &= P^{-1}(VG_2)X_1 = P^{-1}(VG_2)P^{-1}G_1, \\ &\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ X_k &= P^{-1}(VG_k)X_{k-1} = P^{-1}(VG_k)P^{-1}(VG_{k-1})\ldots P^{-1}(VG_2)P^{-1}G_1. \end{aligned} \tag{20.11}$$

Further, using the property 4 and expressing the value $x_k(h)$ by spectral characteristics X_{k-1} and G_k of functions $x_{k-1}(t)$ and $g_k(t)$, respectively, we have

$$x_k(h) = \int_0^h g_k(t)x_{k-1}(t)\mathrm{d}t = G_k^{\mathrm{T}}X_{k-1}.$$

The spectral characteristic X_{k-1} can be represented by Eq. 20.11 as follows:

$$X_{k-1} = P^{-1}(VG_{k-1})\ldots P^{-1}(VG_2)P^{-1}G_1,$$

and

$$x_k(h) = G_k^{\mathrm{T}}P^{-1}(VG_{k-1})\ldots P^{-1}(VG_2)P^{-1}G_1. \tag{20.12}$$

20.4 Using Spectral Form of Mathematical Description to Represent Iterated Stratonovich Stochastic Integrals

Since functions $\{q(i,t)\}_{i=0}^{\infty}$ form the orthonormal basis, their spectral characteristics are columns of the infinite identity matrix E, i.e.,

$$\mathbb{S}[q(i,t)] = E_i, \quad i = 0, 1, 2, \ldots$$

Consider functions $q(i_1,t), q(i_2,t), \ldots, q(i_k,t)$ as input signals of the dynamical system defined by SODE (Eq. 20.8), i.e., $g_l(t) = q(i_l,t)$, $l = 1, 2, \ldots, k$. Then, we obtain relations for expansion coefficients defined by Eq. 20.4: $C_{i_k \ldots i_2 i_1} = x_k(h)$. Thus,

$$C_{i_k \ldots i_2 i_1} = E_{i_k}^{\mathrm{T}} P^{-1} (V E_{i_{k-1}}) \ldots P^{-1} (V E_{i_2}) P^{-1} E_{i_1} \tag{20.13}$$

that follows from Eq. 20.12 with $G_l = E_{i_l}$, $l = 1, 2, \ldots, k$.

The product $V E_{i_l}$ is a section of the three-dimensional infinite matrix V when any index of three indices is fixed at i_l (since V is the symmetric three-dimensional infinite matrix with respect to any pair of indices), and the product $P^{-1} E_{i_l}$ is the section of the infinite matrix P^{-1} at the second fixed index, i.e., its i_l th column. We denote these sections by V_{**i_l} and $P^{-1}_{*i_l}$, respectively. Similarly, the product $E_{i_k}^{\mathrm{T}} P^{-1}$ is the section of the infinite matrix P^{-1} at the first fixed index, i.e., its i_l th row, which we denote $P^{-1}_{i_l *}$. Consequently,

$$C_{i_k \ldots i_2 i_1} = P^{-1}_{i_k *} V_{** i_{k-1}} \ldots P^{-1} V_{** i_2} P^{-1}_{* i_1}.$$

Thus, all expansion coefficients $C_{i_k \ldots i_2 i_1}$ needed for modeling the iterated Stratonovich stochastic integrals are expressed in terms of the infinite matrix P^{-1} and the three-dimensional infinite matrix V.

Iterated stochastic integrals can be expressed by the same matrices P^{-1} and V. It is easy to see that the iterated stochastic integral $I_h^{*(j_1 j_2 \ldots j_k)}$ determined by Eq. 20.1 can be found as a solution to the following system of the Stratonovich stochastic differential equations

$$\begin{aligned}
&\mathrm{d}x_1(t) = \mathrm{d}w_{j_1}(t), \quad x_1(0) = 0,\\
&\mathrm{d}x_2(t) = x_1(t) \circ \mathrm{d}w_{j_2}(t), \quad x_2(0) = 0,\\
&\qquad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&\mathrm{d}x_k(t) = x_{k-1}(t) \circ \mathrm{d}w_{j_k}(t), \quad x_k(0) = 0,
\end{aligned}$$

or the Langevin equations

$$\dot{x}_1(t) = v_{j_1}(t), \quad x_1(0) = 0,$$

$$\dot{x}_2(t) = x_1(t)v_{j_2}(t), \quad x_2(0) = 0,$$
$$\cdots\cdots\cdots\cdots\cdots\cdots$$
$$\dot{x}_k(t) = x_{k-1}(t)v_{j_k}(t), \quad x_k(0) = 0, \tag{20.14}$$

where $j_1, j_2, \ldots, j_k = 1, 2, \ldots, s$ and $v_1(t)$, $v_2(t)$, …, $v_s(t)$ are independent Gaussian white noises corresponding to standard Wiener processes $w_1(t)$, $w_2(t)$, …, $w_s(t)$.

This implies that

$$x_k(t) = \int_0^t \int_0^{\tau_k} \ldots \int_0^{\tau_2} \mathrm{d}w_{j_1}(\tau_1) \circ \mathrm{d}w_{j_2}(\tau_2) \circ \ldots \circ \mathrm{d}w_{j_k}(\tau_k),$$
$$x_k(h) = I_h^{*(j_1 j_2 \ldots j_k)},$$

and the random variables $x_1(h)$, $x_2(h)$, …, $x_{k-1}(h)$ are the iterated Stratonovich stochastic integrals of multiplicities $1, 2, \ldots, k-1$, respectively.

Denote infinite column matrix with entries $\zeta_i^{(j)}$ by $\mathcal{V}_j$. According to [30, 31], $\mathcal{V}_j$ are spectral characteristics of independent Gaussian white noises $v_j(t)$ corresponding to Wiener processes $w_j(t)$, $j = 1, 2, \ldots, s$.

Consider random processes $v_{j_1}(t)$, $v_{j_2}(t)$, …, $v_{j_k}(t)$ as input signals of the dynamical system defined by SODE (Eq. 20.8), i.e., $g_l(t) = v_{j_l}(t)$, $l = 1, 2, \ldots, k$. Then we obtain the representation of the iterated Stratonovich stochastic integrals

$$I_h^{*(j_1 j_2 \ldots j_k)} = \mathcal{V}_{j_k}^{\mathrm{T}} P^{-1}(V\mathcal{V}_{j_{k-1}}) \ldots P^{-1}(V\mathcal{V}_{j_2})P^{-1}\mathcal{V}_{j_1}, \tag{20.15}$$

which follows from Eq. 20.12 with $G_l = \mathcal{V}_{j_l}$, $l = 1, 2, \ldots, k$.

Equation 20.15 can also be obtained by summing up the products of expansion coefficients $C_{i_k \ldots i_2 i_1}$ defined by Eq. 20.13 and random values $\zeta_{i_1}^{(j_1)}\zeta_{i_2}^{(j_2)} \ldots \zeta_{i_k}^{(j_k)}$ taking into account that

$$\mathcal{V}_j = \sum_{i=0}^{\infty} \zeta_i^{(j)} E_i, \quad j = 1, 2, \ldots, s.$$

Equations 20.13 and 20.15 give an exact representation for the expansion coefficients defined by Eq. 20.4 and for the iterated Stratonovich stochastic integral defined by Eq. 20.3. For an approximate representation and modeling, we need to replace infinite matrices in these relations to the corresponding truncated matrices by choosing some truncation order L. Then the column matrices E_i ($i = 0, 1, \ldots, L-1$) and $\mathcal{V}_j$ ($j = 1, 2, \ldots, s$) will be $L \times 1$, the matrix P^{-1} will be $L \times L$, the three-dimensional matrix V will be $L \times L \times L$. According to Eq. 20.15, for the approximate modeling we need to simulate random vectors $\mathcal{V}_1$, $\mathcal{V}_2$, …, $\mathcal{V}_s$ with entries having a standard normal distribution, i.e., we need sL realizations of the standard Gaussian random

variable. This provides simultaneous modeling the set of iterated stochastic integrals of an arbitrary multiplicity.

20.5 Relations for Expansion Coefficients and Iterated Stratonovich Stochastic Integrals

Let us obtain some invariant relations for expansion coefficients defined by Eq. 20.4 corresponding to different multiplicities k. In the simplest case $k = 1$, the expansion coefficients C_{i_1} are integrals of basis functions $q(i_1, t)$ over the interval $[0, h]$:

$$C_{i_1} = \int_0^h q(i_1, t)dt, \quad i_1 = 0, 1, 2, \ldots$$

In fact, these expansion coefficients form the spectral characteristic of the unit step function $1(t)$, i.e., $C_{i_1} = \mathbf{1}_{i_1}$.

For the multiplicity $k = 2$, the matrix formed by expansion coefficients $C_{i_2 i_1}$ coincides with the spectral characteristic P^{-1} of the integration operator: $C_{i_2 i_1} = P^{-1}_{i_2 i_1}$. For these coefficients the following relation holds [15]:

$$C_{i_2 i_1} + C_{i_1 i_2} = C_{i_1} C_{i_2},$$

and this implies that

$$I_h^{*(j_1 j_2)} + I_h^{*(j_2 j_1)} = I_h^{*(j_1)} I_h^{*(j_2)}.$$

The relation for expansion coefficients can be written in the matrix form [10]:

$$P^{-1} + [P^{-1}]^{\mathrm{T}} = \Lambda = \mathbf{1} \cdot \mathbf{1}^{\mathrm{T}},$$

where Λ is the symmetric matrix.

Consider the multiplicity $k = 3$:

$$C_{i_3 i_2 i_1} = E_{i_3}^{\mathrm{T}} P^{-1} (V E_{i_2}) P^{-1} E_{i_1}, \quad C_{i_1 i_2 i_3} = E_{i_1}^{\mathrm{T}} P^{-1} (V E_{i_2}) P^{-1} E_{i_3}.$$

Using properties of the matrix multiplication and transpose, we have

$$\begin{aligned}
[E_{i_3}^{\mathrm{T}} P^{-1} (V E_{i_2}) P^{-1} E_{i_1}]^{\mathrm{T}} &= E_{i_1}^{\mathrm{T}} [P^{-1}]^{\mathrm{T}} (V E_{i_2})^{\mathrm{T}} [P^{-1}]^{\mathrm{T}} E_{i_3} \\
&= E_{i_1}^{\mathrm{T}} (\Lambda - P^{-1}) (V E_{i_2}) (\Lambda - P^{-1}) E_{i_3} \\
&= E_{i_1}^{\mathrm{T}} P^{-1} (V E_{i_2}) P^{-1} E_{i_3} - E_{i_1}^{\mathrm{T}} P^{-1} (V E_{i_2}) \Lambda E_{i_3} \\
&\quad - E_{i_1}^{\mathrm{T}} \Lambda (V E_{i_2}) P^{-1} E_{i_3} + E_{i_1}^{\mathrm{T}} \Lambda (V E_{i_2}) \Lambda E_{i_3}.
\end{aligned}$$

It is easy to see that $\Lambda E_{i_3} = \mathbf{1}_{i_3}\mathbf{1}$. Similarly, $E_{i_1}^{\mathrm{T}}\Lambda = \mathbf{1}_{i_1}\mathbf{1}^{\mathrm{T}}$. Further,

$$(VE_{i_2})\mathbf{1} = (V\mathbf{1})E_{i_2} = EE_{i_2} = E_{i_2},$$
$$\mathbf{1}^{\mathrm{T}}(VE_{i_2}) = [(VE_{i_2})^{\mathrm{T}}\mathbf{1}]^{\mathrm{T}} = [(VE_{i_2})\mathbf{1}]^{\mathrm{T}} = [(V\mathbf{1})E_{i_2}]^{\mathrm{T}} = [EE_{i_2}]^{\mathrm{T}} = E_{i_2}^{\mathrm{T}}.$$

Consequently,

$$E_{i_1}^{\mathrm{T}}P^{-1}(VE_{i_2})\Lambda E_{i_3} = \mathbf{1}_{i_3}E_{i_1}^{\mathrm{T}}P^{-1}E_{i_2} = C_{i_1i_2}C_{i_3},$$
$$E_{i_1}^{\mathrm{T}}\Lambda(VE_{i_2})P^{-1}E_{i_3} = \mathbf{1}_{i_1}E_{i_2}^{\mathrm{T}}P^{-1}E_{i_3} = C_{i_1}C_{i_2i_3},$$
$$E_{i_1}^{\mathrm{T}}\Lambda(VE_{i_2})\Lambda E_{i_3} = \mathbf{1}_{i_1}\mathbf{1}^{\mathrm{T}}(VE_{i_2})\mathbf{1}_{i_3}\mathbf{1} = \mathbf{1}_{i_1}\mathbf{1}_{i_2}\mathbf{1}_{i_3} = C_{i_1}C_{i_2}C_{i_3},$$

i.e.,

$$C_{i_3i_2i_1} = C_{i_1i_2i_3} - C_{i_1i_2}C_{i_3} - C_{i_1}C_{i_2i_3} + C_{i_1}C_{i_2}C_{i_3}.$$

Summing up by i_1, i_2, i_3 the products of random values $\zeta_{i_1}^{(j_1)}\zeta_{i_2}^{(j_2)}\zeta_{i_3}^{(j_3)}$ and both the left-hand side and the right-hand side of the above relation for expansion coefficients, we can write that

$$I_h^{*(j_1j_2j_3)} = I_h^{*(j_3j_2j_1)} - I_h^{*(j_2j_1)}I_h^{*(j_3)} - I_h^{*(j_3j_2)}I_h^{*(j_1)} + I_h^{*(j_3)}I_h^{*(j_2)}I_h^{*(j_1)}.$$

Next, consider the multiplicity $k = 4$:

$$C_{i_4i_3i_2i_1} = E_{i_4}^{\mathrm{T}}P^{-1}(VE_{i_3})P^{-1}(VE_{i_2})P^{-1}E_{i_1},$$
$$C_{i_1i_2i_3i_4} = E_{i_1}^{\mathrm{T}}P^{-1}(VE_{i_2})P^{-1}(VE_{i_3})P^{-1}E_{i_4}.$$

Also using properties of the matrix multiplication and transpose, we have

$$\begin{aligned}
&[E_{i_4}^{\mathrm{T}}P^{-1}(VE_{i_3})P^{-1}(VE_{i_2})P^{-1}E_{i_1}]^{\mathrm{T}}\\
&= E_{i_1}^{\mathrm{T}}[P^{-1}]^{\mathrm{T}}(VE_{i_2})^{\mathrm{T}}[P^{-1}]^{\mathrm{T}}(VE_{i_3})^{\mathrm{T}}[P^{-1}]^{\mathrm{T}}E_{i_4}\\
&= E_{i_1}^{\mathrm{T}}(\Lambda - P^{-1})(VE_{i_2})(\Lambda - P^{-1})(VE_{i_3})(\Lambda - P^{-1})E_{i_4}\\
&= -E_{i_1}^{\mathrm{T}}P^{-1}(VE_{i_2})P^{-1}(VE_{i_3})P^{-1}E_{i_4} + E_{i_1}^{\mathrm{T}}P^{-1}(VE_{i_2})P^{-1}(VE_{i_3})\Lambda E_{i_4}\\
&\quad + E_{i_1}^{\mathrm{T}}P^{-1}(VE_{i_2})\Lambda(VE_{i_3})P^{-1}E_{i_4} + E_{i_1}^{\mathrm{T}}\Lambda(VE_{i_2})P^{-1}(VE_{i_3})P^{-1}E_{i_4}\\
&\quad - E_{i_1}^{\mathrm{T}}P^{-1}(VE_{i_2})\Lambda(VE_{i_3})\Lambda E_{i_4} - E_{i_1}^{\mathrm{T}}\Lambda(VE_{i_2})P^{-1}(VE_{i_3})\Lambda E_{i_4}\\
&\quad - E_{i_1}^{\mathrm{T}}\Lambda(VE_{i_2})\Lambda(VE_{i_3})P^{-1}E_{i_4} + E_{i_1}^{\mathrm{T}}\Lambda(VE_{i_2})\Lambda(VE_{i_3})\Lambda E_{i_4}.
\end{aligned}$$

Applying same properties as well as in the case $k = 3$, we obtain

$$E_{i_1}^{\mathrm{T}}P^{-1}(VE_{i_2})P^{-1}(VE_{i_3})\Lambda E_{i_4} = \mathbf{1}_{i_4}E_{i_1}^{\mathrm{T}}P^{-1}(VE_{i_2})P^{-1}E_{i_3} = C_{i_1i_2i_3}C_{i_4},$$
$$E_{i_1}^{\mathrm{T}}P^{-1}(VE_{i_2})\Lambda(VE_{i_3})P^{-1}E_{i_4} = E_{i_1}^{\mathrm{T}}P^{-1}(VE_{i_2})\mathbf{1}\cdot\mathbf{1}^{\mathrm{T}}(VE_{i_3})P^{-1}E_{i_4}$$

$$
\begin{aligned}
&= E_{i_1}^{\mathrm{T}} P^{-1} E_{i_2} \cdot E_{i_3}^{\mathrm{T}} P^{-1} E_{i_4} = C_{i_1 i_2} C_{i_3 i_4},\\
E_{i_1}^{\mathrm{T}} \Lambda (V E_{i_2}) P^{-1} (V E_{i_3}) P^{-1} E_{i_4} &= \mathbf{1}_{i_1} E_{i_2}^{\mathrm{T}} P^{-1} (V E_{i_3}) P^{-1} E_{i_4} = C_{i_1} C_{i_2 i_3 i_4},\\
E_{i_1}^{\mathrm{T}} P^{-1} (V E_{i_2}) \Lambda (V E_{i_3}) \Lambda E_{i_4} &= \mathbf{1}_{i_4} E_{i_1}^{\mathrm{T}} P^{-1} (V E_{i_2}) E_{i_3}\\
&= \mathbf{1}_{i_3} \mathbf{1}_{i_4} E_{i_1}^{\mathrm{T}} P^{-1} E_{i_2} = C_{i_1 i_2} C_{i_3} C_{i_4},\\
E_{i_1}^{\mathrm{T}} \Lambda (V E_{i_2}) P^{-1} (V E_{i_3}) \Lambda E_{i_4} &= \mathbf{1}_{i_1} \mathbf{1}_{i_4} E_{i_2}^{\mathrm{T}} P^{-1} E_{i_3} = C_{i_1} C_{i_2 i_3} C_{i_4},\\
E_{i_1}^{\mathrm{T}} \Lambda (V E_{i_2}) \Lambda (V E_{i_3}) P^{-1} E_{i_4} &= \mathbf{1}_{i_1} E_{i_2}^{\mathrm{T}} \Lambda (V E_{i_3}) P^{-1} E_{i_4}\\
&= \mathbf{1}_{i_1} \mathbf{1}_{i_1} E_{i_3}^{\mathrm{T}} P^{-1} E_{i_4} = C_{i_1} C_{i_2} C_{i_3 i_4},\\
E_{i_1}^{\mathrm{T}} \Lambda (V E_{i_2}) \Lambda (V E_{i_3}) \Lambda E_{i_4} &= \mathbf{1}_{i_1} \mathbf{1}^{\mathrm{T}} (V E_{i_2}) \mathbf{1} \cdot \mathbf{1}^{\mathrm{T}} (V E_{i_3}) \mathbf{1}_{i_4} \mathbf{1} = C_{i_1} C_{i_2} C_{i_3} C_{i_4},
\end{aligned}
$$

i.e.,

$$
\begin{aligned}
C_{i_4 i_3 i_2 i_1} = &- C_{i_1 i_2 i_3 i_4} + C_{i_1 i_2 i_3} C_{i_4} + C_{i_1 i_2} C_{i_3 i_4} + C_{i_1} C_{i_2 i_3 i_4}\\
&- C_{i_1 i_2} C_{i_3} C_{i_4} - C_{i_1} C_{i_2 i_3} C_{i_4} - C_{i_1} C_{i_2} C_{i_3 i_4} + C_{i_1} C_{i_2} C_{i_3} C_{i_4}.
\end{aligned}
$$

Summing up by i_1, i_2, i_3, i_4 the products of random values $\zeta_{i_1}^{(j_1)} \zeta_{i_2}^{(j_2)} \zeta_{i_3}^{(j_3)} \zeta_{i_4}^{(j_4)}$ and both the left-hand side and the right-hand side of the above relation for expansion coefficients, we get the following relation:

$$
\begin{aligned}
I_h^{*(j_1 j_2 j_3 j_4)} = &- I_h^{*(j_4 j_3 j_2 j_1)} + I_h^{*(j_4)} I_h^{*(j_3 j_2 j_1)} + I_h^{*(j_4 j_3)} I_h^{*(j_2 j_1)} + I_h^{*(j_4 j_3 j_2)} I_h^{*(j_1)}\\
&- I_h^{*(j_4)} I_h^{*(j_3)} I_h^{*(j_2 j_1)} - I_h^{*(j_4)} I_h^{*(j_3 j_2)} I_h^{*(j_1)} - I_h^{*(j_4 j_3)} I_h^{*(j_2)} I_h^{*(j_1)}\\
&+ I_h^{*(j_4)} I_h^{*(j_3)} I_h^{*(j_2)} I_h^{*(j_1)}.
\end{aligned}
$$

Further, consider the multiplicity $k = 5$:

$$
\begin{aligned}
C_{i_5 i_4 i_3 i_2 i_1} &= E_{i_5}^{\mathrm{T}} P^{-1} (V E_{i_4}) P^{-1} (V E_{i_3}) P^{-1} (V E_{i_2}) P^{-1} E_{i_1},\\
C_{i_1 i_2 i_3 i_4 i_5} &= E_{i_1}^{\mathrm{T}} P^{-1} (V E_{i_2}) P^{-1} (V E_{i_3}) P^{-1} (V E_{i_4}) P^{-1} E_{i_5}.
\end{aligned}
$$

Similarly, we can write relations for expansion coefficients as

$$
\begin{aligned}
C_{i_5 i_4 i_3 i_2 i_1} = &- C_{i_1 i_2 i_3 i_4 i_5} - C_{i_1 i_2 i_3 i_4} C_{i_5} - C_{i_1 i_2 i_3} C_{i_4 i_5} - C_{i_1 i_2} C_{i_3 i_4 i_5} - C_{i_1} C_{i_2 i_3 i_4 i_5}\\
&+ C_{i_1 i_2 i_3} C_{i_4} C_{i_5} + C_{i_1} C_{i_2 i_3 i_4} C_{i_5} + C_{i_1} C_{i_2} C_{i_3 i_4 i_5} + C_{i_1 i_2} C_{i_3 i_4} C_{i_5}\\
&+ C_{i_1 i_2} C_{i_3} C_{i_4 i_5} + C_{i_1} C_{i_2 i_3} C_{i_4 i_5} - C_{i_1 i_2} C_{i_3} C_{i_4} C_{i_5} - C_{i_1} C_{i_2 i_3} C_{i_4} C_{i_5}\\
&- C_{i_1} C_{i_2} C_{i_3 i_4} C_{i_5} - C_{i_1} C_{i_2} C_{i_3} C_{i_4 i_5} + C_{i_1} C_{i_2} C_{i_3} C_{i_4} C_{i_5}.
\end{aligned}
$$

Summing up by i_1, i_2, i_3, i_4, i_5 the products of random values $\zeta_{i_1}^{(j_1)} \zeta_{i_2}^{(j_2)} \zeta_{i_3}^{(j_3)} \zeta_{i_4}^{(j_4)} \zeta_{i_5}^{(j_5)}$ and both the left-hand side and the right-hand side of the above relation for expansion coefficients, we obtain:

$$
I_h^{*(j_1 j_2 j_3 j_4 j_5)} = I_h^{*(j_5 j_4 j_3 j_2 j_1)} - I_h^{*(j_5)} I_h^{*(j_4 j_3 j_2 j_1)} - I_h^{*(j_5 j_4)} I_h^{*(j_3 j_2 j_1)}
$$

$$
\begin{aligned}
&- I_h^{*(j_5 j_4 j_3)} I_h^{*(j_2 j_1)} - I_h^{*(j_5 j_4 j_3 j_2)} I_h^{*(j_1)} + I_h^{*(j_5)} I_h^{*(j_4)} I_h^{*(j_3 j_2 j_1)} \\
&+ I_h^{*(j_5)} I_h^{*(j_4 j_3 j_2)} I_h^{*(j_1)} + I_h^{*(j_5 j_4 j_3)} I_h^{*(j_2)} I_h^{*(j_1)} + I_h^{*(j_5)} I_h^{*(j_4 j_3)} I_h^{*(j_2 j_1)} \\
&+ I_h^{*(j_5 j_4)} I_h^{*(j_3)} I_h^{*(j_2 j_1)} + I_h^{*(j_5 j_4)} I_h^{*(j_3 j_2)} I_h^{*(j_1)} - I_h^{*(j_5)} I_h^{*(j_4)} I_h^{*(j_3)} I_h^{*(j_2 j_1)} \\
&- I_h^{*(j_5)} I_h^{*(j_4)} I_h^{*(j_3 j_2)} I_h^{*(j_1)} - I_h^{*(j_5)} I_h^{*(j_4 j_3)} I_h^{*(j_2)} I_h^{*(j_1)} \\
&- I_h^{*(j_5 j_4)} I_h^{*(j_3)} I_h^{*(j_2)} I_h^{*(j_1)} + I_h^{*(j_5)} I_h^{*(j_4)} I_h^{*(j_3)} I_h^{*(j_2)} I_h^{*(j_1)}.
\end{aligned}
$$

Finally, it can be shown that for an arbitrary k:

$$
C_{i_k \ldots i_2 i_1} = \sum_{l=1}^{k} (-1)^{k-l} \sum_{m=1}^{C_{k-1}^{l-1}} M_{lm},
$$

where C_{k-1}^{l-1} is the binomial coefficient [32], and the set of elements M_{lm} for a fixed l is formed by the products of expansion coefficients defined by Eq. 20.4 that are required to represent l iterated stochastic integrals of the total multiplicity k:

$$
\begin{aligned}
&M_{11} = C_{i_1 i_2 \ldots i_k}, \\
&M_{21} = C_{i_1 i_2 \ldots i_{k-1}} C_{i_k}, \quad M_{22} = C_{i_1 i_2 \ldots i_{k-2}} C_{i_{k-1} i_k}, \quad \ldots, \quad M_{2,k-1} = C_{i_1} C_{i_2 i_3 \ldots i_k}, \\
&M_{lm} = C_{i_1 \ldots i_p} C_{i_{p+1} \ldots i_q} \ldots C_{i_{r+1} \ldots i_k}, \quad l = 3, \ldots, k-2, \\
&M_{k-1,1} = C_{i_1 i_2} C_{i_3} \ldots C_{i_k}, \quad M_{k-1,2} = C_{i_1} C_{i_2 i_3} C_{i_4} \ldots C_{i_k}, \quad \ldots, \\
&M_{k-1,k-1} = C_{i_1} \ldots C_{i_{k-2}} C_{i_{k-1} i_k}, \\
&M_{k1} = C_{i_1} C_{i_2} \ldots C_{i_k},
\end{aligned}
$$

where the ordered set of indices $i_1 \ldots i_p i_{p+1} \ldots i_q \ldots i_{r+1} \ldots i_k$ ($1 \leqslant p < q \leqslant r < k$) coincides with $i_1 i_2 \ldots i_k$, i.e., the summation over m is the summation over all possible partitions of the set of indices $i_1 i_2 \ldots i_k$ into l subsets with saving their order.

Therefore, for the iterated Stratonovich stochastic integrals we have:

$$
I_h^{*(j_1 j_2 \ldots j_k)} = \sum_{l=1}^{k} (-1)^{k-l} \sum_{m=1}^{C_{k-1}^{l-1}} M_{lm}^{*},
$$

where

$$
\begin{aligned}
&M_{11}^{*} = I_h^{*(j_k \ldots j_2 j_1)}, \\
&M_{21}^{*} = I_h^{*(j_k)} I_h^{*(j_{k-1} \ldots j_2 j_1)}, \quad M_{22}^{*} = I_h^{*(j_k j_{k-1})} I_h^{*(j_{k-2} \ldots j_2 j_1)}, \quad \ldots, \\
&M_{2,k-1}^{*} = I_h^{*(j_k \ldots j_3 j_2)} I_h^{*(j_1)}, \\
&M_{lm}^{*} = I_h^{*(j_k \ldots j_{r+1})} \ldots I_h^{*(j_q \ldots j_{p+1})} I_h^{*(j_p \ldots j_1)}, \quad l = 3, \ldots, k-2, \\
&M_{k-1,1}^{*} = I_h^{*(j_k)} \ldots I_h^{*(j_3)} I_h^{*(j_2 j_1)}, \quad M_{k-1,2}^{*} = I_h^{*(j_k)} \ldots I_h^{*(j_4)} I_h^{*(j_3 j_2)} I_h^{*(j_1)}, \quad \ldots,
\end{aligned}
$$

$$M^*_{k-1,k-1} = I_h^{*(j_k j_{k-1})} I_h^{*(j_{k-2})} \dots I_h^{*(j_1)},$$
$$M^*_{k1} = I_h^{*(j_k)} \dots I_h^{*(j_2)} I_h^{*(j_1)},$$

and the ordered set of indices $j_k \dots j_{r+1} \dots j_q \dots j_{p+1} j_p \dots j_1$ $(1 \leqslant p < q \leqslant r < k)$ coincides with $j_k \dots j_2 j_1$, i.e., the summation over m is the summation over all possible partitions of the set of indices $j_k \dots j_2 j_1$ into l subsets with saving their order.

For complete orthonormal systems such as the Legendre polynomials, trigonometric functions, the Walsh functions, and the Haar functions with the standard numeration, the function $1(t)$ differs from the basic function $q(0, t)$ by the numerical coefficient, i.e., $1(t) = \sqrt{h}\, q(0, t)$, and for $h = 1$ they are equal. Therefore,

$$\mathbf{1} = [\sqrt{h} \;\; 0 \;\; 0 \;\; \dots]^{\mathrm{T}},$$

i.e., $C_{i_1} = \sqrt{h}$ for $i_1 = 0$ and $C_{i_1} = 0$ for $i_1 > 0$. This simplifies above relations for expansion coefficients:

$$\begin{aligned}
&C_{i_2 i_1} = -C_{i_1 i_2}, \quad C_{i_1 i_1} = 0, \quad i_1, i_2 > 0;\\
&C_{i_3 i_2 i_1} = C_{i_1 i_2 i_3}, \quad i_1, i_3 > 0;\\
&C_{i_4 i_3 i_2 i_1} = -C_{i_1 i_2 i_3 i_4} + C_{i_1 i_2} C_{i_3 i_4}, \quad C_{i_4 i_3 i_1 i_1} = -C_{i_1 i_1 i_3 i_4},\\
&C_{i_4 i_4 i_2 i_1} = -C_{i_1 i_2 i_4 i_4}, \quad i_1, i_4 > 0;\\
&C_{i_5 i_4 i_3 i_2 i_1} = C_{i_1 i_2 i_3 i_4 i_5} - C_{i_1 i_2 i_3} C_{i_4 i_5} - C_{i_1 i_2} C_{i_3 i_4 i_5},\\
&C_{i_5 i_5 i_3 i_1 i_1} = C_{i_1 i_1 i_3 i_5 i_5}, \quad i_1, i_3, i_5 > 0,
\end{aligned}$$

and so on.

In the general case $M_{lm} = 0$ if $\lfloor k/l \rfloor = 1$ and $i_1, i_2, \dots, i_k > 0$, where $\lfloor \cdot \rfloor$ is the floor function.

Obtained invariant relations can reduce computational costs for the calculation of expansion coefficients and for modeling the iterated Stratonovich stochastic integrals. In fact, if the coefficient $C_{i_1 i_2 \dots i_k}$ has been calculated, then the coefficient $C_{i_k \dots i_2 i_1}$ has also been calculated. Similarly, if the iterated Stratonovich stochastic integral $I_h^{*(j_k \dots j_2 j_1)}$ has been modeled, then we can model the iterated Stratonovich stochastic integral $I_h^{*(j_1 j_2 \dots j_k)}$ by $I_h^{*(j_k \dots j_2 j_1)}$ and integrals of multiplicity less then k.

20.6 Tensor Representation

Definitions and properties of the spectral transform listed in Sect. 20.3 can be extended to functions of several variables. We will use some notations from [26, 28] for spectral characteristics of functions of several variables. Then

$$\mathbb{S}[K(t_1, t_2, \ldots, t_k)] = C_{(k)},$$

i.e., $C_{(k)}$ is the spectral characteristic of the function $K(t_1, t_2, \ldots, t_k)$ defined with respect to the basis $\{q(i_1, t_1)q(i_2, t_2)\ldots q(i_k, t_k)\}_{i_1,i_2,\ldots,i_k=0}^{\infty}$, or the k-dimensional hypercolumn matrix with elements $C_{i_1 i_2 \ldots i_k}$. The hypercolumn matrix with entries $\hat{C}_{i_1 i_2 \ldots i_k} = C_{i_k \ldots i_2 i_1}$ will be denoted $\hat{C}_{(k)}$, it is related to $C_{(k)}$ by the "mirror" reorder of indices.

Denote

$$\mathcal{V}^{(j_1 j_2 \ldots j_k)} = \mathcal{V}_{j_1} \otimes \mathcal{V}_{j_2} \otimes \ldots \otimes \mathcal{V}_{j_k},$$

where $\mathcal{V}_j$ are spectral characteristics of independent Gaussian white noises $v_j(t)$ defined earlier ($j = 1, 2, \ldots, s$), and $\otimes$ means the tensor multiplication of multidimensional matrices [26, 28].

Then Eq. 20.3 can be written as

$$I_h^{*(j_1 j_2 \ldots j_k)} = \hat{C}_{(k)}^{\mathrm{T}} \mathcal{V}^{(j_1 j_2 \ldots j_k)}, \tag{20.16}$$

where $[\cdot]^{\mathrm{T}}$ means the transition from the hypercolumn matrix to the hyperrow matrix [26, 28].

Relations that connect expansion coefficients $C_{i_k \ldots i_2 i_1}$ for different multiplicities k (see Sect. 20.5) can be written in the matrix form using the definition of hypercolumn matrices $C_{(k)}$ and $\hat{C}_{(k)}$ as well as the tensor multiplication of multidimensional matrices:

$$\begin{aligned}
\hat{C}_{(2)} &= -C_{(2)} + C_{(1)} \otimes C_{(1)},\\
\hat{C}_{(3)} &= C_{(3)} - C_{(2)} \otimes C_{(1)} - C_{(1)} \otimes C_{(2)} + C_{(1)} \otimes C_{(1)} \otimes C_{(1)},\\
\hat{C}_{(4)} &= -C_{(4)} + C_{(3)} \otimes C_{(1)} + C_{(2)} \otimes C_{(2)} + C_{(1)} \otimes C_{(3)}\\
&\quad - C_{(2)} \otimes C_{(1)} \otimes C_{(1)} - C_{(1)} \otimes C_{(2)} \otimes C_{(1)} - C_{(1)} \otimes C_{(1)} \otimes C_{(2)}\\
&\quad + C_{(1)} \otimes C_{(1)} \otimes C_{(1)} \otimes C_{(1)},\\
\hat{C}_{(5)} &= C_{(5)} - C_{(4)} \otimes C_{(1)} - C_{(3)} \otimes C_{(2)} - C_{(2)} \otimes C_{(3)} - C_{(1)} \otimes C_{(4)}\\
&\quad + C_{(3)} \otimes C_{(1)} \otimes C_{(1)} + C_{(1)} \otimes C_{(3)} \otimes C_{(1)} + C_{(1)} \otimes C_{(1)} \otimes C_{(3)}\\
&\quad + C_{(2)} \otimes C_{(2)} \otimes C_{(1)} + C_{(2)} \otimes C_{(1)} \otimes C_{(2)} + C_{(1)} \otimes C_{(2)} \otimes C_{(2)}\\
&\quad - C_{(2)} \otimes C_{(1)} \otimes C_{(1)} \otimes C_{(1)} - C_{(1)} \otimes C_{(2)} \otimes C_{(1)} \otimes C_{(1)}\\
&\quad - C_{(1)} \otimes C_{(1)} \otimes C_{(2)} \otimes C_{(1)} - C_{(1)} \otimes C_{(1)} \otimes C_{(1)} \otimes C_{(2)}\\
&\quad + C_{(1)} \otimes C_{(1)} \otimes C_{(1)} \otimes C_{(1)} \otimes C_{(1)}.
\end{aligned}$$

For an arbitrary k we have:

$$\hat{C}_{(k)} = \sum_{l=1}^{k} (-1)^{k-l} \sum_{\substack{k_1,k_2,\ldots,k_l \geqslant 1 \\ k_1+k_2+\ldots+k_l=k}} \bigotimes_{\alpha=1}^{l} C_{(k_\alpha)}.$$

The tensor representation is convenient to implement algorithms for modeling iterated stochastic integrals using computer algebra systems or matrix algebra subroutine packages.

20.7 Conclusions

In this chapter, the spectral form of mathematical description for the representation of the iterated Stratonovich stochastic integrals of an arbitrary multiplicity is applied. For this purpose, we need to calculate both the spectral characteristic of the integration operator and the spectral characteristic of the multiplier. These spectral characteristics may be defined with respect to an arbitrary complete orthonormal system for the representation and modeling. Obtained invariant relations can reduce computational costs for the calculation of expansion coefficients and for modeling the iterated Stratonovich stochastic integrals. For expansion coefficients, the tensor representation is also obtained.

References

1. Milstein, G.N.: Approximate integration of stochastic differential equations. Theor. Prob. Appl. **19**(3), 583–588 (1974)
2. Milstein, G.N.: Numerical Integration of Stochastic Differential Equations. Kluwer Academic Publ, Dordrecht (1995)
3. Milshtein, G.N., Tretyakov, M.V.: Stochastic Numerics for Mathematical Physics. Springer, Berlin (2004)
4. Kloeden, P.E., Platen, E.: Numerical Solution of Stochastic Differential Equations. Springer, Berlin (1992)
5. Kuznetsov, D.F.: A method of expansion and approximation of repeated stochastic Stratonovich integrals based on multiple Fourier series on full ortonormal systems. Diff. Eqn. Control Process. (1), 18–77 (in Russian) (1997)
6. Prigarin, S.M., Belov, S.M.: One Application of Series Expansions of Wiener Process. Preprint 1107. ICM & MG Publ., Novosibirsk, Russia (in Russian) (1998)
7. Wiktorsson, M.: Joint characteristic function and simultaneous simulation of iterated Ito integrals for multiple independent Brownian motions. Ann. Appl. Probab. **11**(2), 470–487 (2001)
8. Ryden, T., Wiktorsson, M.: On the simulation of iterated Itô integrals. Stoch. Process Their Appl. **91**(1), 151–168 (2001)
9. Tang, X., Xiao, A.: Asymptotically optimal approximation of some stochastic integrals and its applications to the strong second-order methods. Adv. Comput. Math. **45**(3), 813–846 (2019)
10. Rybakov, K.A.: Applying spectral form of mathematical description for representation of iterated stochastic integrals. Diff. Eqn. Control Process. (4), 1–31 (in Russian) (2019)

11. Rybakov, K.A.: Application of Walsh series to represent iterated Stratonovich stochastic integrals. IOP Conf. Ser.: Mater. Sci. Eng. **927**, 012080 (2020)
12. Artemiev, S.S., Averina, T.A.: Numerical Analysis of Systems of Ordinary and Stochastic Differential Equations. VSP, Utrecht (1997)
13. Kuznetsov, D.F.: Approximation of Multiple Ito and Stratonovich Stochastic Integrals. Lambert, Saarbrücken (2012)
14. Han, X., Kloeden, P.E.: Random Ordinary Differential Equations and their Numerical Solution. Springer, Singapore (2017)
15. Kuznetsov, D.F.: Strong approximation of iterated Itô and Stratonovich stochastic integrals based on generalized multiple Fourier series. Application to numerical solution of Itô SDEs and Semilinear SPDEs. Diff. Eqn. Control Process. **4** (2020)
16. Karachanskaya, E.V.: Programmed control with probability 1 for stochastic dynamical systems. J. Math. Sci. **248**, 67–79 (2020)
17. Rybakov, K.A.: Solving approximately an optimal nonlinear filtering problem for stochastic differential systems by statistical modeling. Numer. Anal. Appl. **6**(4), 324–336 (2013)
18. Rybakov, K.A.: Robust Duncan-Mortensen-Zakai equation for non-stationary stochastic systems. In: IEEE Int. Multi-Conference on Engineering, Computer and Information Sciences, pp. 151–154 (2017)
19. Rudenko, E.: Algorithms and programs of suboptimal nonlinear filtering for Markov processes. AIP Conf. Proc. **2181**, 020017 (2019)
20. Bortakovskii, A.S., Nemychenkov, G.I.: Suboptimal on average satellite attitude control in the presence of discrete inaccurate measurements. J. Comput. Sys. Sci. Int. **57**(2), 197–207 (2018)
21. Khalina, A.S., Khrustalev, M.M.: Effect of displacement of optimal control in stabilization problems for quasi-linear diffusion-type stochastic systems. J. Comput. Syst. Sc. Int. **58**(2), 159–166 (2019)
22. Petukhov, V.G., Ivanyukhin, A.V.: Woo Sang Wook: Joint optimization of control and main trajectory and design parameters of an interplanetary spacecraft with an electric propulsion system. Cosmic Res. **57**(3), 188–203 (2019)
23. Davtyan, L.G., Panteleev, A.V.: Method of parametric optimization of nonlinear continuous systems of joint estimation and control. J. Comput. Syst. Sci. Int. **58**(3), 360–373 (2019)
24. Kuznetsov, D.F.: Application of the method of approximation of iterated stochastic Ito integrals based on generalized multiple Fourier series to the high-order strong numerical methods for non-commutative semilinear stochastic partial differential equations. Diff. Eqn. Control Process. (3), 18–62 (2019)
25. Solodovnikov, V.V., Semenov, V.V., Peschel, M., Nedo, D.: Design of Control Systems on Digital Computers: Spectral and Interpolational Methods. Mashinostroenie, Moscow (in Russian), Verlag Technik, Berlin (in German) (1979)
26. Rybakov, K.A., Sotskova, I.L.: Spectral method for analysis of switching diffusions. IEEE Trans. Autom. Control **52**(7), 1320–1325 (2007)
27. Panteleev, A.V., Rybakov, K.A.: Analyzing nonlinear stochastic control systems in the class of generalized characteristic functions. Autom. Remote Control **72**(2), 393–404 (2011)
28. Baghdasaryan, G.Y., Mikilyan, M.A., Panteleev, A.V., Rybakov, K.A.: Spectral method for analysis of diffusions and jump diffusions. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Computational Mechanics and Numerical Simulation. Smart Innovation, Systems and Technologies, vol. 173, pp. 293–314. Springer, Singapore (2020)
29. Kuznetsov, D.F.: Expansion of iterated Stratonovich stochastic integrals based on generalized multiple Fourier series. Ufa Math. J. **11**(4), 49–77 (2019)
30. Rybakov, K.A.: Modeling and analysis of output processes of linear continuous stochastic systems based on orthogonal expansions of random functions. J. Comput. Syst. Sci. Int. **59**(3), 322–337 (2020)

31. Rybakov, K.A.: Spectral method of analysis and optimal estimation in linear stochastic systems. Int. J. Model. Simul. Sci. Comput. **11**(3), 2050022 (2020)
32. Korn, G.A., Korn, T.M.: Mathematical Handbook for Scientists and Engineers. Dover Publ, New York (2000)

Part V
Information Technologies

Chapter 21
Fractal Analysis and Programming of Elastic Systems Using Container-Component Model

Alexander S. Semenov

Abstract The analysis and design of distributed algorithms is one of the main reasons to use fractal programming. Its aims are to represent the distributed algorithm as an "elastic object" that transforms dynamically at runtime. The use of container-component model provides the following advantages: the ability to select automatically a distributed configuration, building a visual model of the elastic computing organization, and evaluating its effectiveness. Container-component model is integrated with the box-counting fractal analysis method and fractal control based on dynamic sampling of the workload. The example of fractal analysis and programming of the distributed gradient ascent algorithm is given.

21.1 Introduction

The analysis and design of distributed algorithms is one of the main reasons to use fractal programming [1, 2]. Its aims are to represent the distributed algorithm as an "elastic object" that transforms dynamically at runtime using strategy planning model and production rules. The distributed configuration of the algorithm unfolds at the beginning of execution and folds at the end of execution. The use of container-component model (CCM) for the analysis and programming of elastic algorithms provides the following advantages: the ability to automatically select a distributed configuration of the algorithm execution, building a visual model of the elastic computing organization, and evaluation of its effectiveness.

Fractal analysis, design, and programming of complex elastic systems are integrated with the mathematical methods of the fractal analysis on the base of CCM. Physically, a distributed system consists of a number of nodes (autonomous computers) interconnected by a network. The nodes may be computers, physical servers, agents, virtual machines, containers, or other entities that can connect to the

A. S. Semenov (✉)
Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, Moscow 125993, Russian Federation
e-mail: semenov_alex@yahoo.com

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_21

network, have local memory, and communicate by passing messages [3, 4]. The basic models of a distributed system are the message passing and shared memory models [5, 6]. The message is the unit of communication of a distributed system. Distributed algorithms are designed to run on different computers of the network. Parts of an algorithm run concurrently and independently, each with a limited amount of information [7]. There are two problems: how to describe adequately a distributed algorithm and how to prove its properties. Designing the appropriate distribution [8] and formation control [9] is a general problem, especially for multi-agent systems with complex dynamics. A distributed algorithm is adequately described if it takes into account property of the computing environment. Graphs [10] and adjusted version of Petri nets [11, 12] are used to model the distributed system and distributed algorithms.

The design of efficient distributed algorithms is very important for analyzing big data. Three properties of big data such as velocity, volume, and variety must be considered. Technically, distributed algorithms to process big data are designed for a specific application and mainly based on the MapReduce framework [13]. The framework implements two functions Map and Reduce. Map function divides the input data into data partitions that constitute key-value pairs. Each partition is assigned to a unique compute node. Nodes outputs are one or more intermediate key-value pairs. The framework collects all the intermediate key-value pairs, sorts, and groups them by key. The Reduce function aggregates the values associated with the key according to a predefined program and stores all the output key-value pairs in a file. Hadoop is an open-source project written in Java that implements MapReduce framework [14]. It is possible to write some MapReduce jobs in Python and then run them in Hadoop Streaming that operates like the pipes in Linux. Hadoop jobs are running on Amazon Web Services that provides on-demand cloud computing platforms. This is fundamental support to big data [15] that are unprecedented content for Digital Earth [16].

In cloud computing systems, an on-demand network model is used to provide access to shared pool of configurable IT resources. One of the biggest features of cloud computing is their scalability and elasticity [17]. The elasticity of an application is a measure of its transformation that depends on fluctuating demands [18]. The approach presented in the article is an integral part of the analysis, design, and fractal programming of elastic systems.

One way to characterize the elastic object is to compute the fractal dimension. In the chapter, fractal analysis of the elastic objects based on the box-counting fractal dimension is introduced. CCM is integrated with the box-counting fractal analysis method. The model has a high level of abstraction, operates with container objects, simulates different levels of the control granularity, and makes predictions about behavior of transformation. The integration of the model makes possible to introduce a fractal control based on a dynamic sampling of the workload. The fractal analysis and programming aim to optimize suitable workflow structure of the elastic object at runtime and its ability to runtime elasticity, which must be modeled and be built into the system at design time. The example of the fractal analysis and programming of the distributed gradient ascent algorithm using integrated CCM is considered.

The chapter is organized as follows. In Sect. 21.2, the box-counting analysis of the capacity curve of the elastic system is considered. In Sect. 21.3, the definition of CCM is given. The model is integrated with the box-counting method. The fractal control of the elastic object is introduced. In Sect. 21.4, the container-component fractal analysis of the distributed gradient ascent algorithm is given. Section 21.5 concludes the chapter.

21.2 The Box-Counting Fractal Analysis of the Capacity Curve

The core of the elastic computing is the workload and capacity. The assumption is made that workload on the system equals to capacity of the system. The fractal analysis base on box counting of the capacity curve of the elastic computer system is represented in this section.

Hereinafter, the capacity curve of the elastic object is discussed in Sect. 21.2.1. The analysis of the capacity curve is given in Sect. 21.2.2.

21.2.1 The Capacity Curve of the Elastic Object

The elastic concepts characterize a service. The incremental, decremental, and iterative nature of elastic object directly impacts of it construction and object hierarchies in the design time of a complex software system. Due to this, the capacity of the services needs to be incremented or decremented by IT resources with characterized curve. Let workload on the system equals its capacity.

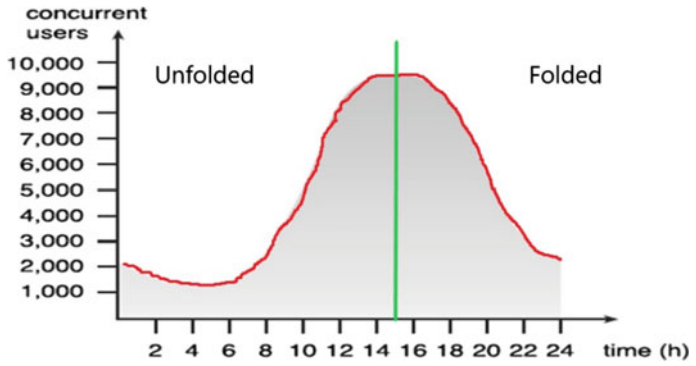


Fig. 21.1 Fluctuating demand of the users during the day is increasing and decreasing

The transformation of the elastic object at runtime is shown in Fig. 21.1 by a curve, where the unfolded object on the left side of the curve and folded one on the right side of the curve are presented.

21.2.2 The Analysis of the Capacity Curve

There are many fractal dimensions introduced in mathematical literature, e.g., [19–21]. Here, the capacity curve is analyzed by the box-counting dimension. The ideas of the method are the following:

1. Determine curve via boxes covered it.
2. Cover the maximum element of a curve by boxes.
3. Scale only the boxes, do not scale the curve.

Definition 1 Let N be the number of boxes calculated by function n: $(\delta, \varepsilon) \rightarrow N$, where ε is a grid (a box size) that contains at least one element of the curve δ. If

$$d_\delta = \lim_{\varepsilon \to 0} \frac{\log(n(\delta, \varepsilon))}{\log(1/\varepsilon)} \tag{21.1}$$

exists, then the limit is called the box-counting dimension (Minkowski–Bouligand dimension) of δ and is denoted by d_δ.

The computation of the dimension d_δ begins by selecting a set of box sizes. For each value of ε, the minimal number (n) of boxes of size ε needed to cover δ is determined. The box counting of the capacity curve in dependence on the scaling ratio of the ε-grid boxes is shown in Fig. 21.2 and Table 21.1. For Fig. 21.2a, Eq. 21.1 is rewritten by Eq. 21.2.

$$d_\delta = \lim_{\varepsilon \to 0} \frac{\log(7)}{\log(1/\varepsilon)} = 1 \tag{21.2}$$

The analysis of Table 21.1 shows that left side of a capacity curve needs more boxes than right side. Such as one of the axes is a time axis, a conclusion has been done that a process of unfolded object needs more than one fold.

But the box-counting analysis does not decide a problem how to construct and control the elastic object. These problems are decided by the integration of a box-counting method with CCM.

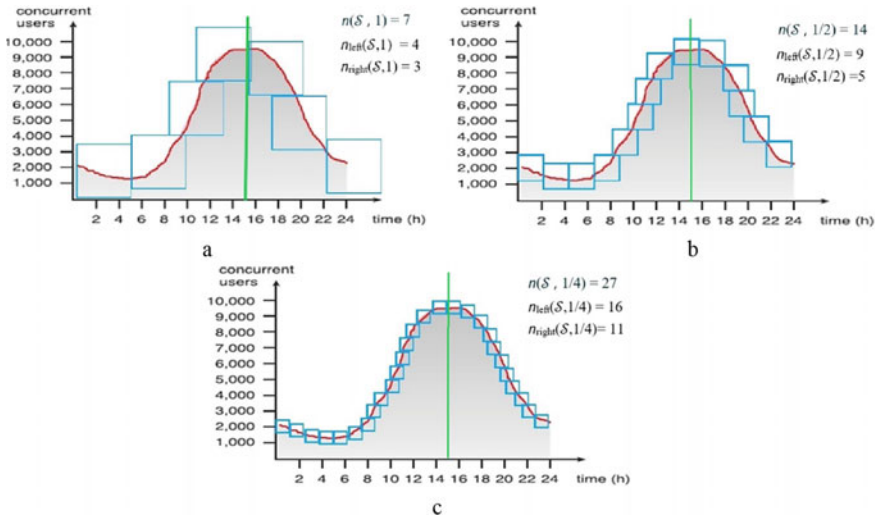


Fig. 21.2 Box counting of the capacity curve: **a** initial grid, **b** grid with ratio 1/2, **c** grid with ratio 1/4

Table 21.1 Box counting of the capacity curve

Figure 21.2	$n(\delta, \varepsilon)$	$n_{\text{left}}(\delta, \varepsilon)$	$n_{\text{right}}(\delta, \varepsilon)$
a	$n(\delta, 1) = 7$	$n_{\text{left}}(\delta,1) = 4$	$n_{\text{right}}(\delta, 1) = 3$
b	$n(\delta, 1/2) = 14$	$n_{\text{left}}(\delta, 1/2) = 9$	$n_{\text{right}}(\delta, 1/2) = 5$
c	$n(\delta, 1/4) = 27$	$n_{\text{left}}(\delta, 1/4) = 16$	$n_{\text{right}}(\delta, 1/4) = 11$

21.3 Container-Component Fractal Analysis and Fractal Control

The definition of CCM as a fractal construction is considered in Sect. 21.3.1. Then in Sect. 21.3.2, this model is integrated with the box-counting method. This provides a possibility to introduce a fractal control of elastic object which is adjusted to the box-counting analysis.

21.3.1 Container-Component Model

Fundamental concept of a model is an elastic container object. A container object has the following properties: contains other containers, contains IT resources (data, functions, and methods), manages the storage space of its elements and provides access to them, and moves from one computing environment to another. Containers can be used very effectively to identify non-serving and non-interactive programs that

simply accept input and return a result. Let each container run on its own server. Server does not share global memory and communicates exclusively through messaging. The model algorithm is constructed as follows.

Let upper indexes R (Receive) and S (Send) be the inputs and outputs for the container e_0, respectively, then a container program is defined by Eq. 21.3.

$$E = {}^{R}\{e_0\}^{S} \tag{21.3}$$

A distributed program E consists of a set of n asynchronous container programs ${}^{R}\{e_0\}^{S}, {}^{R}\{e_1\}^{S}, {}^{R}\{e_2\}^{S}, \ldots, {}^{R}\{e_n\}^{S} \in E$, which interact through containers transmitted over the network, and a number of containers n vary over time in accordance with the requirements.

Components fulfill also a role of communication programs between container programs. The component program is designated as ${}^{R}\{c\}^{S}$ and cannot be divided.

Depending on each specific case, the operation to receive (R) and to send (S) can be redefined, which is indicated by the appropriate upper index for the curly brace. If there is no index, then an operation is not applied.

Definition 2 Container-component model $E = f^{n}$ (${}^{R}\{e_0\}^{S}$, ${}^{R}\{c\}^{S}$, Δ), where f^{n}: $E \rightarrow E$ is a recursive mapping of a *set* E in itself by Δ operations, $n = 0, 1, \ldots, k$ is a step of mapping, ${}^{R}\{e_0\}^{S}$ is an initial container program, ${}^{R}\{c\}^{S}$ is a component program, $\Delta = \{\equiv(1/r, N), \downarrow, n{+}{+}, n{-}\}$ is an ordered set of uniquely invertible operations, $\equiv(1/r, N)$ ${}^{R}\{e\}^{S}$ is an operation of prototyping container program ${}^{R}\{e\}^{S}$ with parameters, r is a scaling ratio (it means that container program is divided into r containers and component programs), N is a number of self-similar container programs ${}^{R}\{e\}^{S}$ after division, $1/r \neq N$; $\downarrow: E' \rightarrow e$ is an insert set E' in the container e, $n{+}{+}$ is an increment operation, $n = n + 1$, $n-$ is a decrement operation, $n = n - 1$.

Components obtained by the prototyping operation are used also to simulate message passing between neighboring container programs and the container program from which they are prototyped. Figure 21.3 shows an intercommunication between the initial program ${}^{R}\{e_0\}^{S}$ and subprograms ${}^{R}\{e_1^1\}^{S}$ and ${}^{R}\{e_3^1\}^{S}$ if $r = 1/3$ and $n = 1$. Graphically, container is depicted by a rectangle. Nested containers are depicted by the nested rectangles. Component is depicted by a gray rectangle.

21.3.2 Container-Component Model Integrated with the Box-Counting Method

Let each container $e \in E$ be a box of square size and the side of the box equal to $u(t)$, where $u(t)$ is the control signal sent to the elastic system, see Fig. 21.4. Initial box-container and $u_0(t)$ is corresponded to $e_0 \in E$. Graphically, box-container is depicted by a rectangle. Nested containers are depicted by nested rectangles. The component is depicted by yellow rectangle. The capacity curve is analyzed by the box-container counting dimension. The ideas of the method are different from a box counting:

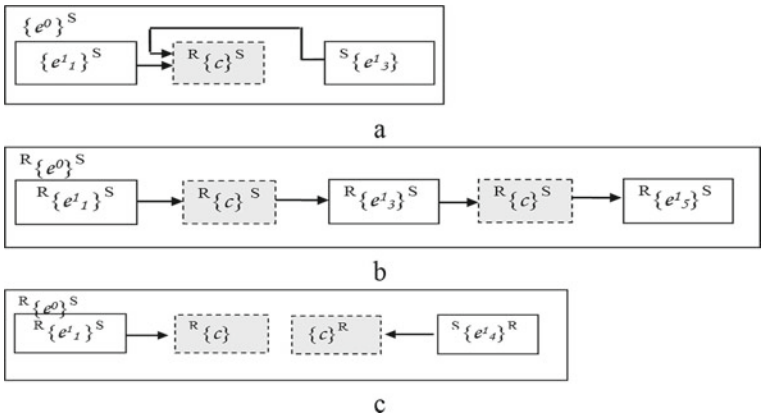


Fig. 21.3 CCM pattern in the first iteration step, $n = 1$: **a** $1/r = 3, N = 2$, **b** $1/r = 5, N = 3$, **c** $1/r = 4, N = 2$

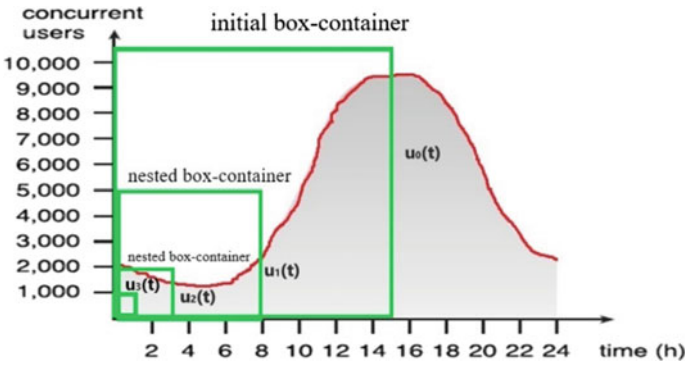


Fig. 21.4 Integration of CCM with a box-counting method

1. Determine curve via boxes covered it.
2. Cover the element of a curve along one of the axe by boxes.
3. Define the sizes of maximal and minimal boxes.
4. The repeatable boxes on one axe do not take part in counting: They are skipped and marked by cross.

Definition 3 Let N be the number of boxes calculated by function n: $(\delta, _{\max}) \to N$, where $\varepsilon_{\max}$ is a maximal box size, and $\varepsilon_{\min}$ is a minimal box size that contains at least one element of the curve δ and ordered along axes. If

$$d_u = \lim_{\varepsilon_{\max} \to \varepsilon_{\min}} \frac{\log(n(\delta, \varepsilon_{\max}))}{\log(1/\varepsilon_{\max})} \tag{21.4}$$

exists, then the limit is called the box-counting dimension of δ and is denoted by d_u.

The box-container counting of the capacity curve in dependence on the scaling ratio of the $\varepsilon_{\max}$, $\varepsilon_{\min}$ grid boxes is shown in Fig. 21.5 and Table 21.2.

The sizes of maximal and minimal boxes are corresponded to the maximal and minimal control signals. The signals $u_0(t)$, $u_1(t)$, $u_2(t)$, and $u_3(t)$ are grained scaling of the elastic object control.

The value of function $u(t)$ is defined for every value of time t. It is a function of a continuous independent variable. *Discrete-time signals* are defined only at discrete times that form a discrete set of values of the independent variable. This is usually done by sampling [22] a continuous-time signal at isolated, equally spaced points in time T. The result is a sequence of numbers defined by $u[m]$ where m is an integer $\{0, 1, 2, 3, \ldots.\}$. In this chapter, continuous independent variables are enclosed in parentheses (), and discrete-independent variables are enclosed in square brackets [].

Definition 4 A discrete-time system is a system that transforms a discrete-time input signal $u[m]$ into a discrete-time output signal $y[m]$.

A discrete-time signal $u[m]$ takes values from a finite set of K integers $\{v_1, v_2, \ldots, v_K\}$. This value is equal to the number of containers n required to control the system.

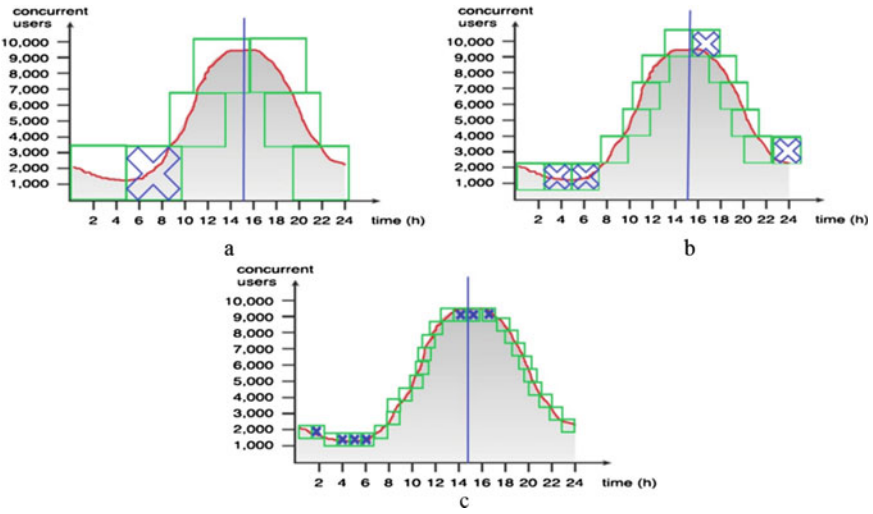


Fig. 21.5 Box-container counting of the capacity curve: **a** initial grid, **b** grid with ratio 1/2, **c** grid with ratio 1/4

Table 21.2 Container-box counting of the capacity curve

Figure 21.5	$n(\delta, \varepsilon_{\max})$	$n_{\text{left}}(\delta, \varepsilon_{\max})$	$n_{\text{right}}(\delta, \varepsilon_{\max})$
a	$n(\delta, 1) = 6$	$n_{\text{left}}(\delta, 1) = 3$	$n_{\text{right}}(\delta, 1) = 3$
b	$n(\delta, 1/2) = 10$	$n_{\text{left}}(\delta, 1/2) = 7$	$n_{\text{right}}(\delta, 1/2) = 3$
c	$n(\delta, 1/4) = 20$	$n_{\text{left}}(\delta, 1/4) = 11$	$n_{\text{right}}(\delta, 1/4) = 9$

A discrete-time system is denoted symbolically by

$$y[m] = f\{u[m]\}, \tag{21.5}$$

where f denotes CCM characterizing the system.

In a box-counting method, it takes that a curve is given. Let a part of a curve named sample is formed dynamically at a given time interval T.

Definition 5 Let N be the number of boxes calculated dynamically by function n: ($\delta[m \cdot T]$, $\varepsilon_{\max}$) $\rightarrow$ N, where $\delta\,[m \cdot T]$ is a curve sample calculated by Eq. 21.4, then control signals $u[m]$ is denoted by Eq. 21.6.

$$u[m] = u[m \cdot T] = \begin{cases} n_{\text{left}}(\delta[m \cdot T], \varepsilon_{\max}) & \text{if } m > m-1 \\ n_{\text{right}}(\delta[m \cdot T], \varepsilon_{\max}) & \text{if } m < m-1 \\ 0 & \text{if } m \leq 0 \end{cases} \tag{21.6}$$

Figure 21.6 shows a dynamic sampling of the capacity curve at a given time interval $T = 1$, and the capacity of one container equals to 1000 req/s.

The granularity characterizes a number of containers with a defined capacity that should be prototyped in response to the control signal. One control signal of the elastic system with maximal granularity $u_0[m]$ needs $T = 21$, $n = 11$ number of containers for transformation (unfolded and folded). The granularity $u_1[m]$ needs $T = 8$, $n = 5$ containers. The granularity $u_2[m]$ needs $T = 3$, $n = 2$ containers. The granularity $u_3[m]$ needs $T = 1$, $n = 1$.

On the other hand, a number of containers C for CCM can be calculated by Eq. 21.7.

$$C = \frac{N^{n+1} - 1}{N - 1} \tag{21.7}$$

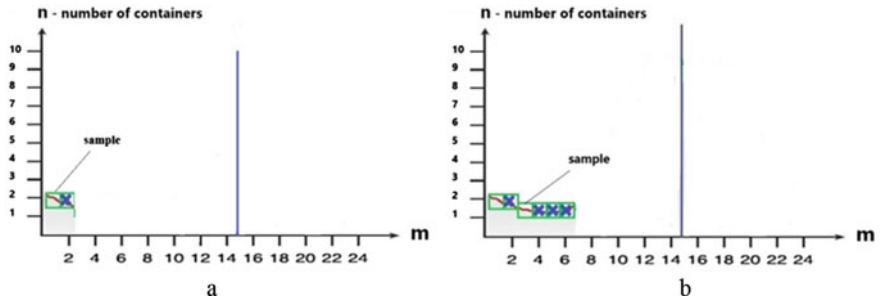


Fig. 21.6 Dynamic sampling of the capacity curve: **a** sample $m = 1$, [1], $n = 2$, and sample $m = 2$, δ[2] does not affect the control, **b** next sample $m = 3$, δ[3], $n = 1$, and samples δ[4], δ[5], δ[6] do not affect the control

Equality between Eqs. 21.6 and 21.7 is held by Eq. 21.8.

$$u[m] = \frac{N^{n+1} - 1}{N - 1} \tag{21.8}$$

Then a number of steps k of the elastic model is calculated by Eq. 21.9.

$$k = \left[\log_N((N-1) \cdot u[m] + 1)\right] \tag{21.9}$$

CCM for the pattern $1/r = 3$, $N = 2$ (see Fig. 21.3a) is defined by Eq. 21.10.

$$\begin{aligned} & f^{\mathrm{n}}\left(E^0 = \{e_0\}^{\mathrm{S}}, {}^{\mathrm{R}}\{c\}^{\mathrm{S}},\right. \\ & [n++, \forall\{e\}^{\mathrm{S}} \in E^{n-1}\{e\}^{\mathrm{S}} \downarrow (E^n = \equiv (3,2)\{e\}^{S}) | 0 < n \leq k], \\ & \left.[n--, \forall\{e\}^{\mathrm{S}} \in E^{n-1}\{e\}^{\mathrm{S}} \downarrow (E^n = \equiv^{-1} (3,2)\{e\}^{S}) | 0 < n \geq k]\right) \end{aligned} \tag{21.10}$$

21.4 The Fractal Analysis of the Distributed Gradient Ascent Algorithm

In this section, the analysis based on integrated CCM of gradient ascent algorithm is presented. At the beginning, a gradient ascent algorithm [23] is considered in Sect. 21.4.1, and after that, its distributed version based on integrated CCM is introduced in Sect. 21.4.2.

21.4.1 Gradient Ascent for Big Data Ordered in Time

Let big data be ordered at discrete-time T with equal intervals τ. Figure 21.7 roughly illustrates this. The idea is to identify a slope and move it up. This method does not

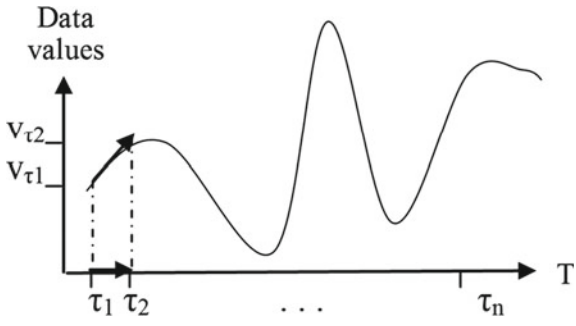


Fig. 21.7 Gradient ascent for big data ordered in time

require to compute or even know $f(x)$, but it computes the slope, that is, $v = v_{\tau 2} - v_{\tau 1}$, where $v_{\tau 2} - v_{\tau 1}$ is a slop. If a slope is positive, then $\max(v)$ will increase. If a slope is negative, then $\max(v)$ will decrease.

The idea is to identify a slope between the neighboring points. The algorithm is mentioned below.

Algorithm 1. Gradient Ascent for Big Data ordered in time

1. V is a random initial vector ordered at discrete-time $T = [0,..., \tau_n]$
   $\max(v_\tau) = 0$ is an initial value of random vector max/min $v_\tau \in V$
2. for each τ in T
3. if $\max(v_\tau) < v_\tau$
4. $\max(v_\tau) = v_\tau$
5. end for each
6. return $\max(v_\tau)$

The algorithm runs until found $\max(v_\tau)$ or $\min(v_\tau)$.

21.4.2 Distributed Algorithm: Gradient Ascent for Big Data Ordered in Time

Our objective is to construct the distributed CCM that captures a relationship between the computing time m of Algorithm 1 (A1) and capacity c. For solution of this issue, the following assumptions are made:

- Let vector V has about 2000 M (Millions) values and changeable during some period of time, $V = 2000$ M.
- Let m be the computation time, and c be the capacity of the system that handles by $u[m]$, see Eq. 21.6.
- Let a server be encapsulated in container e_0. The capacity of the server $c = 2$ MWIPS (Millions of Whetstone Instruction Per Second [24, 25]) is a constant, and each server encapsulated in container has the same capacity.
- Algorithm A1 consists from three operations in one loop ($w = 3$): For each τ in $T = [0, \ldots, \tau_n]$, condition operator is less then "<", and assignment operator is "=".

Thus, the computation time m of Algorithm 1 with $u[m]$ containers is presented by Eq. 21.1.

$$m = V/(c * u[m]/w) \tag{21.11}$$

Figure 21.8 shows a container-component tree, where the containers are marked by capacity c.

For the computation time m, vector V needs $u[m] = V \cdot w/(m \cdot c) = 2000\,\text{M} \cdot 3/(2\,\text{MWIPS} \cdot u[m])$ containers. The calculation of algorithm can be planned

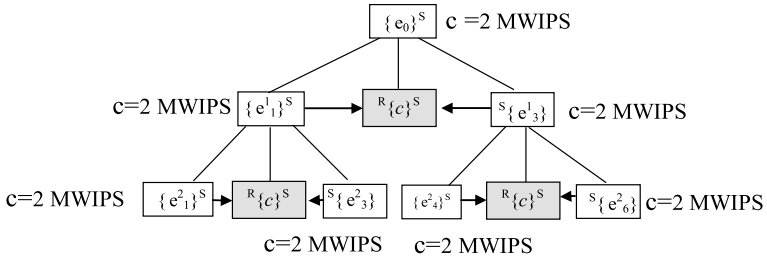


Fig. 21.8 Component-container model of the distributed gradient ascent algorithm: $1/r = 3$, $N = 2$, $m \approx 429$ s, $n = 2$

by setting a capacity curve. Let $m \approx 429$ s and $u[m] = 7$. Then a number of steps k is calculated by Eq. 21.9.

There is no loss of generality in assuming that each container E runs A1 with next rule for each container program in accordance with operation of prototyping $\equiv (1/r, N)$:

$$T = [0, \ldots, \tau_n] = \equiv (1/r, N)[0, \ldots, \tau_n]. \tag{21.12}$$

CCM for gradient ascent algorithm and pattern $1/r = 3, N = 2$ (see Fig. 21.3a) is presented by Eq. 21.13.

$$\begin{aligned} E \;=\; & f^{\mathrm{n}}\,(E^0 = \{e_0\}^{\mathrm{S}} = \mathrm{A1}, {}^{\mathrm{R}}\{c\}^{\mathrm{S}}, \\ & [n++, \forall\{e\}^{\mathrm{S}} \in E^{n-1}\{e\}^{\mathrm{S}} \downarrow (E^n = \equiv (3,2)\{e\}^{S}) | 0 < n \leq k], \\ & [n--, \forall\{e\}^{\mathrm{S}} \in E^{n-1}\{e\}^{\mathrm{S}} \downarrow (E^n = \equiv^{-1} (3,2)\{e\}^{S}) | 0 < n \geq k]) \end{aligned} \tag{21.13}$$

When elastic component-container tree is folded, the result will be in the initial container component. One of the potential problems is that elasticity takes time.

21.5 Conclusions

The contribution of the chapter is the application of elastic CCM to design the distributed algorithms. As a result of this current study, several conclusions can be drawn:

- Distributed algorithm can be presented as an "elastic object" which is transformed dynamically at runtime.
- CCM provides the following advantages: the ability to automatically select a distributed configuration of the data processing organization, building a visual model of elastic computing organization, and evaluation of its effectiveness.

- Integration CCM with the box-counting fractal analysis method and with fractal control based on dynamic sampling of the workload makes it possible to introduce fractal control of elastic system.
- Scheduling the distributed algorithm computation is implemented by setting a capacity curve.

Future work concerns the problems of analysis and design different distributed algorithms, patterns of organizing the data processing, scheduling the distributed algorithm computation, and investigation of elastic system with fractal control.

References

1. Semenov, A.S.: Essentials of fractal programming. In: Jain, L.C., Favorskaya, M.N., Nikitin, I.S., Reviznikov, D.L. (eds.) Advances in Theory and Practice of Computational Mechanics: Proceedings of the 21st International Conference on Computational Mechanics and Modern Applied Software Systems, SIST, vol. 173, pp. 373–386. Springer, Singapore. (2020)
2. Semenov, A.S.: Prototype based programming with fractal algebra. AIP Conf. Proc. **2181**, 020009 (2019)
3. Tel, G.: Introduction to distributed algorithms. Cambridge University Press, Cambridge (2000)
4. Santoro, N.: Design and Analysis of Distributed Algorithms. Wiley Inc., New Jersey (2007)
5. Raynal, M.: Distributed Algorithms for Message-Passing Systems. Springer, Berlin Heidelberg (2013)
6. Raynal, M.: Fault-Tolerant Message-Passing Distributed Systems. An Algorithmic Approach. Springer, Berlin (2018)
7. Lynch, N.A.: Distributed Algorithms. Morgan Kaufmann Publishers, Inc., San Francisco, California (1996)
8. Zhongkui, L, Zhisheng, D.: Cooperative Control of Multi-Agent Systems a Consensus Region Approach. Taylor & Francis Group (2015)
9. Rastgoftar, H.: Continuum Deformation of Multi-Agent Systems. Springer International Publishing AG (2016)
10. Erciyes, K.: Distributed Graph Algorithms for Computer Networks. Springer, London (2013)
11. Reisig, W.: Elements of Distributed Algorithms: Modeling and Analysis with Petri Nets. Springer Science & Business Media (2013)
12. Semenov, A.S.: Fractal Petri nets. In: 4th International Conference on Control, Decision and Information Technologies. Barcelona, Spain, pp. 1174–1179 (2017)
13. Oussous, A., Benjelloun, F., Lahcen, A., Belfkih, S.: Big data technologies: a survey. J. King Saud Univ. Comput. Inf. Sci. **30**(4), 431–448 (2018)
14. Harrington, P.: Machine Learning in Action. Manning Publications (2013)
15. Yanga, C., Huangb, Q., Lic, Z., Liua, K., Hua, F.: Big Data and cloud computing: innovation opportunities and challenges. Int. J. Digit. Earth **10**(1), 13–53 (2017)
16. Guo, H., Goodchild, M., Annoni, A. (eds.): Manual of Digital Earth. Springer, Singapore (2016)
17. Herbst, N. R., Kounev, S., Reussner, R.: Elasticity in cloud computing: What it is, and what it is not. In: 10th International Conference on Autonomic Computing San Jose, CA, pp. 23–27 (2013)
18. Becker, S., Brataas, G., Lehrig, S. (eds.): Engineering Scalable, Elastic, and Cost-Efficient Cloud Computing Applications. The CloudScale Method. Springer, Cham (2017)
19. Peitgen, H., Jurgens, H., Saupe, D.: Chaos and Fractals. New Frontiers of Science. Springer New York, Inc, New York (2004)
20. Crownover, R.: Introduction to Fractals and Chaos. Jones and Bartlett Publishers, Inc. (1995)

21. Rosenberg, E.A.: Survey of Fractal Dimensions of Networks. Springer Briefs in Computer Science. Springer, Cham (2018)
22. Manolakis, D.G., Ingle, V.K.: Applied Digital Signal Processing. Theory and practice. Cambridge University Press, Cambridge (2011)
23. Luke, S.: Essentials of metaheuristics, 2nd edn. Online Version 2.2 (2015)
24. Core i7-5960X extreme edition review: Intel's overdue desktop 8-core is here. https://techgage.com/article/core-i7-5960x-extreme-edition-review-intels-overdue-desktop-8-core-is-here/. Last accessed 25 Aug 2020
25. AMD threadripper 3990X review: A 64-core multithreaded beast unleashed. https://hothardware.com/reviews/amd-ryzen-threadripper-3990x-cpu-review. Last accessed 25 Aug 2020

Chapter 22
On the Modeling of the University Education Processes in the Information Technology

Vladimir N. Lukin and Lev N. Chernyshov

Abstract The problem of training qualified professionals in the field of information technology is very acute. At the level of the ministry and universities, measures are being taken to stimulate teachers, but the result is poorly felt. The reason is an exaggerated assessment of scientific activity to the detriment of education. The model proposed in the work shows the inefficiency of this approach: it reduces the level of control influence in negative feedback and does not lead to the formation of a stable learning process. In addition, feedback from the student is not taken into account, and the teacher does not seek to raise his/her level. But for successful work in the rapidly changing field of information technology, it is impossible to achieve quality without this. Thus, the teacher needs to find time for self-training. It is natural to use software tools to support the educational process. Existing tools are difficult to use. In addition, the teacher for various reasons hardly understands them. Given the high entry threshold, the authors propose simple and accessible software tools that allow one to free up teacher time for effective student training. The proposed solution does not pretend to completeness, but it makes it possible to form control materials, conduct control measures of different levels, take into account the attendance of classes, the dynamics of the educational process, and maintain interaction with a group of students. It is natural to spend the free time to improve your own skills.

V. N. Lukin · L. N. Chernyshov (✉)
Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, 125993 Moscow, Russian Federation
e-mail: levchernvn@gmail.com

V. N. Lukin
e-mail: lukinvn@list.ru

L. N. Chernyshov
Financial University under the Government of the Russian Federation, 49, Leningradsky prosp., 125993 Moscow, Russian Federation

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_22

22.1 Introduction

Students studying at a university should upon graduation become engineers or researchers possessing high qualifications. The training of highly qualified specialists is the main purpose of the university education.

We consider only information technology. There is a high need for such specialists due to the constant appearance of new technological solutions and large number of development areas. Before talking about optimizing the process of learning at a university, we determine who should be considered as a high-quality graduate. We assume that these are the specialists necessary for industry (employers). Authoritative experts do not get tired of talking about this problem, for example, [1]. Thus, it is necessary to be guided by the needs of the employer, with the prospect of 3–5 years ahead. For information technology specialties, this is a very serious challenge. Moreover, if we talk about specific needs, it is unrealistic: during this time, technologies are very likely to be different. The university cannot quickly respond to modern requirements: technology should have stable features. But it is also impossible to ignore the trend toward information technologies [2].

We assume the requirements for the university, teacher, and student by higher authorities are aimed at educating a high-quality specialist, and not at an effective assessment of highly effective practice [3]. Of course, this is a very strong assumption, but quite acceptable. However, to simplify the task, we abstract from the requirements for the university, as well as, from the requirements for the student, and restrict ourselves to the role of the teacher, who interacts with the student to achieve the necessary quality. Let us pay attention: we solve the problem of teacher workload, and not optimize the educational process.

Section 22.2 provides a model for achieving the required quality of training under specified conditions. Section 22.3 discusses software tools that should simplify the work of a teacher, as well as, independently organize an information space for working with students and colleagues. Section 22.4 describes simple software solutions that allow a particular teacher to simplify the management of the educational process, including the conduct of control measures. Section 22.5 concludes the chapter.

22.2 Models of the Educational Process at the University

We define the parameters that characterize a high-quality teacher, his/her costs to correspond to the necessary level of these parameters, and his/her resource. By resource we will mean time.

Note that a high-quality teacher is characterized mainly by one parameter: high-quality (sought-after) graduates (this parameter is a key for the university). More precisely, this parameter reflects the qualification and the ability to transfer knowledge and experience. This parameter is not easy to measure, although an indirect

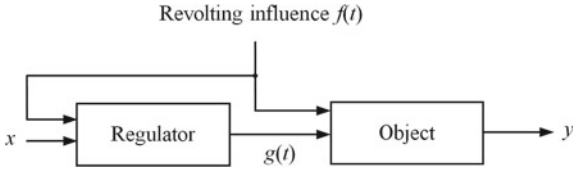


Fig. 22.1 Control with disturbance compensation

estimation is possible. However, modern evaluations are performed according to other performance criteria. They are combined into a system of indicators that should encourage teachers to work effectively. We call it "an effective contract with a teacher" and outline its goal—to optimize the educational process.

We will build the simplest one-dimensional model of educational process management, which is aimed at achieving the stated goal. In the initial approach, this is an open system, in which the control device sets the control action without receiving information about the state of the system. This is how the administrative management system was built. But for the training of the specialist that the employer needs, this option is unacceptable: the ease of management does not compensate for its poor quality. A somewhat more complex variant is related to the response to a disturbance (Fig. 22.1).

Here, the role of the regulator is performed by the teacher, the object is a graduate, the entrance (x) is the rules governing the activity of the teacher, and exit (y) is the quality of the graduate. Disturbing impact is the information about the deviations of the object from the established boundaries (e.g., low entrance level of the student, program change, changing the requirements for the graduate, and new performance criteria). In the effective contract, just this option is implemented. Here, control is carried out according to criteria that are practically unrelated to the educational process, and the relationship between the controlling effect and its result is not visible. Such a model is simple and reliable, but it has a low quality of management.

A model that uses the negative feedback looks much more attractive. For some driving effect $g(t)$, which may be a training course taught by the teacher, the output $y(t)$ of the object (e.g., student's knowledge) is evaluated and a control error is determined: the difference between the required and the current output value a $\varepsilon(t) = g(t) - y(t)$. In case of a nonzero error, the value $\varepsilon(t)$ is supplied to the input of the regulator, which forms a control effect to obtain ideally $\varepsilon(t) = 0$. Thus, a closed loop is formed (Fig. 22.2). This makes the system more resistant to accidental parameter changes.

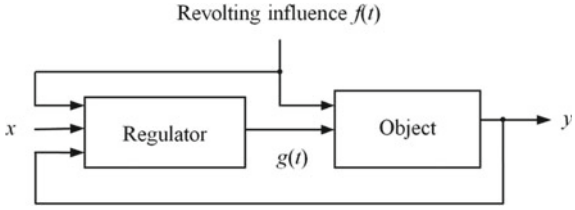


Fig. 22.2 Feedback control

Within the semester, such a model can be considered as continuous if the "temperature" of the occupations is regularly monitored, and if noticeable deviations ($\varepsilon(t)$) from what the teacher considers the norm ($y(t)$), make appropriate adjustments. Here, the output is the current training of the student. If we consider the output as the readiness of the graduate, the model becomes discrete. The quality of the graduate is compared with the desirable quality, and at $\varepsilon(t) > 0$ conclusions about some correction of the course are drawn. In the case where a graduate specializes in information technology, a regulatory error is always there because the natural lag in the curriculum from real needs. This amount can be reduced, moreover, made insignificant if the teacher has the time and desire to constantly modernize the course. Otherwise, the feedback breaks, and we return to the old model, releasing specialists whom the employer will have to train in the workplace.

Thus, we conclude that the model from one-dimensional becomes at least two-dimensional: $y = (y_1, y_2)$, where the components are the qualifications of the student (y_1) and the teacher (y_2). If y_1 stabilizes, then y_2 is an increasing value. If the regulator detects an insufficient growth (a decrease compared to the increasing requirements of the employer), it should stimulate the teacher's further training.

What could be the consequences of a regulator error in this case? If the gain $\varepsilon_2(t) = g_2(t) - y_2(t)$ is excessive, that is, the feedback will be positive, the teacher will spend his/her resources to the detriment of the educational process without a significant positive result. Otherwise, the course degrades, it will not become modern, and the graduate will cease to be interesting for the employer.

We considered the student as a management object that does not affect the course of the educational process. In fact, in reality this is not the case. The student (reasonable) strives to be interesting for the employer and makes efforts for this purpose. If from his/her point of view, the teacher satisfies this requirement, he/her tries to get to win him/her for a diploma, otherwise he/her ceases to be an object of management (walking, working on the side), and the university cannot provide his/her full-fledged quality. Now the model (Fig. 22.2) is interpreted as "inverted": the object is a teacher, the regulator is a student. In fact, it is necessary to consider the superposition of these models, reflecting the interaction complexity of the elements. A simplified version is provided in Fig. 22.3.

Thus, it becomes obvious that in order to achive the desired quality the information technology teacher must constantly improve his/her skills. The first thing that can

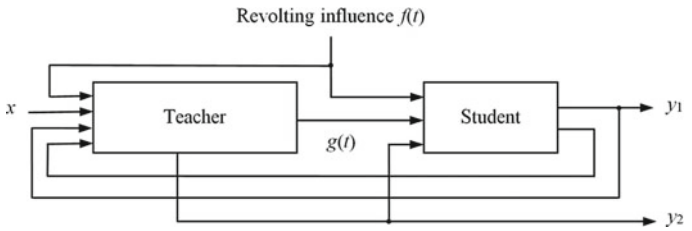


Fig. 22.3 Model of student–teacher's interaction

come to mind is an advanced training system that is available in every university. But in fact, this option does not always work: the set of courses usually does not include current topics, and if it does, then the program for the interpretive teacher is no longer interesting, as he/her already knows everything. Fortunately, in our specialty, there is a wonderful way of improvement: a participation in real program projects. The authors have been lucky enough to do such a work for many years, and they can appreciate the undeniable benefit of it in the educational process, especially if students or recent graduates participate in the projects. But such a path is not possible for everyone. Then a self-education remains, within the framework of which you can carry out useful projects "for yourself". We will come back to this point in the sequel. And finally, there remain such tools as conferences and seminars, at which participants usually meet with interesting thoughts and suggestions.

Thus, the necessary condition for the preparation of quality graduates is a good idea of what is needed at the exit ($g(t)$), the ability to create a package of disciplines that allows one to form the required output competence ($y(t)$), and the ability to do this.

For simplicity, we will assume that the teacher's qualifications are good enough to create and support training courses in the current state. The question of how to obtain the information about the real exit requirements was discussed at the beginning of the article. It remains to determine the possibility of forming competence, and above all—where to get time.

We consider the range of work of graduate staffs, who are interested in educating high-quality graduates (Table 22.1).

Let us consider, first of all, to the main load, the educational one. In Moscow Aviation Institute (MAI) (National University), the annual load is 1534 h, of which 870 h are per audience, 664 h remain. Close estimates are given in [4]. What are these 664 h used for?

First of all, the items A.1 and A.2, as well as, A.5 and A.6, without this are impossible. A.10 could not be done, but required. A.4 would like to use for promising students, but you have to spend on working with twins. Works B, C, and D have to keep within what remained from this rest completely.

Now, if we look at which of these elements are included in an effective contract with the teacher, simply, for which they earn bonus points, we will see that the study works, except for the work with graduate students (A.9), are not evaluated in any way. And in assessing the work with graduate students, only their protection is appreciated. Works B and C are evaluated mainly, which make up the input of our model. It is easy to see that they influence the real production management of qualitative graduates and qualitative teachers poorly, except unless, C.5 point, that is, the model is almost opened.

Thus, it is necessary to cut out time for "effective" work, that is, to be considered at a university not an empty place from the point of view of the managers. We do not forget that we need time to correct the model control errors in order to train in-demand specialists. And the first thing that comes to mind, in addition to intensifying the work (work for wear), is to automate everything that you can, fortunately, specialty gives such an opportunity.

Table 22.1 List of works from the point of view of the teacher

№.	Works	Academic load	Points	Possible to automate
A. Study's work				
1	Preparation for classes			
2	Development of IT disciplines			
3	Training	*		
4	Individual work with students			
5	Preparation of examinations (questions, tickets)			*
6	Preparation of control materials: tasks, tests			*
7	Performance of tests, exams, coursework	*		*
8	Work with diploma students	*		*
9	Work with graduate students	*	*	
10	Reporting on academic work			*
B. Methodic work				
1	Preparation of methodological materials: lecture notes, teaching aids, etc			
2	Preparation of training programs		*	
3	Preparation and publication of textbooks and manuals		*	
C. Scientific work				
1	Prepare and publish monographs		*	
2	Publication of articles		*	
3	Reports at conferences		*	
4	Research work		*	

(continued)

Table 22.1 (continued)

№.	Works	Academic load	Points	Possible to automate
5	Program projects		*	
6	Research work of students		*	
D. Scills' development				
1	Defense of a thesis		*	
2	As part of advanced training programmers			
3	Self-preparation			

*indicates the types of work related to the training load, considered in an effective contract and subject to automation

It is immediately clear that only a part of the educational work can be automated. Neither methodological, nor scientific work, nor advanced training due to this is significantly simplified. Let us see how you can reduce manual work for some types of study work, thereby gaining additional time for other work.

22.3 Software Support for the Learning Process

This section discusses software tools that are designed to simplify the learning process. However, the teacher is far from always comfortable working with them. Simple and accessible tools are offered that make it easier to work with a group of students. Section 22.3.1 demonstrates how to generate unique variants of home, control, and laboratory work. Section 22.3.2 shows how to use cloud platforms to fully manage a group of students in a distance learning environment. When conducting an exam or sweep, paper tickets are usually used, which in the remote version do not work. In Sect. 22.3.3, there are the proposed option with electronic documents, as well as, coursework and thesis. Sections 22.3.4 and 22.3.5 show how to organize work performance monitoring along with generation and distribution of options. Section 22.3.6 concerns the organization of communication. Regular e-mail is not always convenient, so it is proposed to use the option to work with the group.

Software tools in the educational process are used at all levels: university, faculty, department, and teachers. The level of the university is needed for the schedule, the level of the faculty for working with student groups, the level of the department—the training load, the level of the teacher—methodological materials, evaluation, and control. It is assumed that the main software will be a university level system that covers the needs of other levels. Let 'us see what teachers can do.

Software tools that are primarily designed to simplify the learning process include Moodle [5], Easy LMS [6], WizIQ Virtual Classroom [7], and NextThought [8].

Systems that allow online training, such as the same Moodle or Populi [9], provided invaluable assistance in forced remote training during quarantine. However, the use of such systems, as practice has shown, does not particularly save the teacher's time, sometimes just the opposite. More useful in this sense are the tools related to test preparation and evaluation.

As for complex solutions at the university level, it is worth mentioning the domestic development of University Information and Analytical System (UIAS) of the Moscow Aviation Institute based on the 1C System [10]. Its goal is a general information environment where data on employees, students, courses, programs of disciplines, schedules, etc., are stored. True, there is no support for the remote educational process, unlike Moodle, which is widely used in the world. The OpenEduCat [11] is also of interest, but it focuses on administrative functions, as in Collegix from Aptron [12]. For a detailed overview, see [13].

It would seem that the abundance of existing solutions gives the teacher the opportunity to choose the most suitable one that will facilitate his/her work, reduce fatigue, and increase labor productivity. However, here we are faced with an unexpected phenomenon: almost none of the teachers voluntarily use these funds. Factors such as entry threshold, complexity of use, insufficient efficiency are affected. The first factor means that a complex, multifunctional product requires time to study, which is already too short. The second factor is that the development of the product is not usually done by teachers, and its authors do not feel the needs of the user. In addition, both age and the level of qualification of faculty members in the field of information technology affect [14]. The third factor is related to the fact that developers often make a product of interest to the university administration (it is easier to introduce it) or overload it with extra functions that are often poorly connected [15]. As a result, work is not facilitated, but difficult. The teacher of "information" disciplines in this sense is in a better position: he/her is more freely guided in many software products and their features. In addition, he/her can find solutions that can do without bulky applications.

From general resources, the teacher usually uses a schedule, student lists, performance tools, etc. In addition, he/her usually uses office products (Word, Excel) and e-mail. However, there is still a lot of routine work that takes a considerable time. Teachers in computer disciplines are faced with the need to use software products that act as a subject. There is an additional load associated with the preparation of programming tasks and their verification: last year's backlog is not always possible to use. Note that tasks are most often individual, so the tendency to increase groups of students leads to an additional increase in a workload.

All the software necessary for the teacher can be conditionally divided into two groups: the one used periodically and the other used daily. The first group includes, for example, funds for the preparation of Working Programs of Disciplines (WPDs) or an individual plan. They must be common to all, they must comply with the standards, and therefore they are subject to serious requirements. The software of the second group is used in the educational process on a daily basis, it is often created and accompanied by the teachers themselves. Here are some examples.

Testing is a widely used training technology, including in IT training. Here, a significant problem is the compilation of test tasks. It is solved in different ways: from attracting students to using test task generators.

IT disciplines require specialized tests such as DataBases (DB) [16]. Their peculiarity is not comparison with the reference report, but verification by executing some code on the server (program or SQL query). Testing programs are included as part of various distance learning systems, but there are also autonomous programs. Test tasks are usually developed directly by the teachers themselves. There are, of course, test generation systems. Sometimes simple tools based on templates are also suitable. One of these is described in the following section.

22.3.1 Homework, Monitoring, and Laboratory Work

The main problem for the teacher is the development of work options of this type. It is desirable that the options not be repeated not only in one group of students, but also for different groups. If you change WPD and add new topics, you must update the tasks and/or add new ones.

Automation of tasks, the text of which can be voluminous, is significantly difficult. If in any cases it is possible to parameterize the text, the generation can be done by a simple program. Give an example. In the following program on the JavaScript, the first object contains the text of a job that has parameters—constructs of type "%par%". They must be replaced with the values of certain arrays. The values are selected randomly. The result is the desired number of jobs that differ in parameter values. Another template produces a similar program with other objects. Pseudo-code is shown in Fig. 22.4.

22.3.2 Distribution of Options

For both the distance learning and the regular learning, it is convenient to use cloud platforms. For each student, a folder is created, for example, on a Google disk, in which prepared versions are placed. Here, students also place the results of work. This is easy to automate with Google scripts. Their source code of which is posted on github [17]. All information is displayed in the Google table, which serves as a check. If the result is a program on JavaScript or Python, it starts in one click (for Python, the Google Colab extension is used). The view of the table for a group is shown in Fig. 22.5. In the cells with underlined words, there are the links, where you can go to student folders, open task conditions, tables with assessments, visits, and options. The job column displays the date the student downloads his/her decision. At the same time, the date is a reference, for which you can open a job. Use the Changes menu item to update the table.

```
mtemp = [
"Write HTML-code with the functions on the JavaScript. ",
"A display form has buttons (%pos%)and a selection item (%pos%).",
"%typ1% is selected and %typ2% is written to the selection item.",
"By result button %typ3%."
]
mpos = ["on the top upper right corner", "on the lower right corner "]
mtyp = [
["cinema", " movie list and session price", " is the movie with the
lowest ticket price"],["theater", " list of performances and date", "
there is a performance with the earliest date"]
function getRandomInt(min, max){ // random number in the range [min,max]
… }
function get(m){ // accidental value from m … }
function mix(m){ // mix m … }
f2 = fso.CreateTextFile(fout) // to create the output file
for (j=0; j<NV; j++){  // NV - number of options
        for (ks=0; ks<mtemp.length; ks++) {
                s  = mtemp[ks]  // 1
                ms = s.split("%") // text % code %  text % code %  text …
                k = ms.length
                if (k>1){ // есть %kod%
                        i0 = 0
                        for (var i=1; i<k; i+=2){ // by codes in line
                                if (ms[i]=='k')        ms[i]='2'
                                else if (ms[i]=='pos'){
                                        if (i0==0) m1 = mix(mpos) // mix
                                        ms[i]= m1[i0];
                                        i0++
                                }else if (ms[i]=='typ1'){ //
                                        k1 = mtyp.length
                                        k2 = getRandomInt(0, k1-1)
                                        ms[i]   = mtyp[k2][0]
                                        ms[i+2] = mtyp[k2][1]
                                        ms[i+4] = mtyp[k2][2]
                                }else if       (ms[i]=='out'){
                            // . . .
                         }
                }
                 ss = ""; for(i=0;i<k;i++) ss+=ms[i] //assembly of a line
                } else ss =s
                f2.WriteLine(ss)
        }
        f2.WriteLine("")
}
f2.Close()
```

Fig. 22.4 Pseudo-code of program written on the JavaScript

PI19	Works	Correspondence				
Questions	Options					
Attendance	Points					
			Files	Task-1	point	Task-2
Abramov	abrv@gmail.com	14.04.2020	14	20.4	6	27.4
Alekseev	alex@gmai.com	24.03.2020	8	24.5	7	

Fig. 22.5 Print screen of Google-table “Table of a single group”

Discipline	Groups	Group Lists	Questions	Studwork	Visit assessment	Tasks variant
TDWP	PI3-1	PI3-1 studs	05.03.2020	studwork	visit	variant
	PI3-2	PI3-2 studs	14.03.2020	studwork	visit	variant
MTP	PI19	PI3-1 studs	01.03.2020	studwork	visit	variant

Fig. 22.6 Print screen of Google-table "Table of all the groups"

This table is for the instructor. The student have access to the visit tables, estimates, and options for viewing. According to the menu item "Put scores", the scores from the columns with scores are transferred to the rating table.

It is convenient for a teacher conducting classes in several groups to have a common table (Fig. 22.6). You can access the links in the table cells to the appropriate folders or tables. In the column "Questions", the name of the link is the modification date of the document with questions, which allow the teacher to quickly respond to students' questions.

The program attached to the table has the function to create a folder and a table structure for the new group. This is done in several steps. A folder is created for a new discipline, and a link to it is indicated in the first column. Then, the second column after the discipline name indicates the group name, and the New Group Folder menu item creates a group table with an empty student list. After the list of students with their accounts is filled in, the subfolders for students' work are created and other tables are filled in.

The teacher can create a shortcut on the desktop with a link to this table and, when it is opened, immediately see whether the documents with questions were updated: if the date is updated, the cell with the date is highlighted. Discipline folders contain materials, tasks, examples, etc. Due to the fact that the same documents can belong to different folders in the file system of a Google disk, the posted materials can be made available to different groups. To do this, one can use a special table with a script that provides such an access.

The teacher is comfortable with the "Group Table" to work with the group, and a particular student is primarily interested in his/her own information. To do this, there is a "Student Table" (Fig. 22.7). And if in other disciplines teachers will keep the same tables, the student can display his/her information for several disciplines at once. By copying this table into his/her folder and setting up a data sample for himself, the student will receive a kind of a diary.

List of Task				
Section	Task	Execution Date	Point	Penalty
1	task-14	08.06.2020	8	1
2	task-15	15.06.2020	10	0
+ ≡	Students ▼	Tasks ▼	Visit of classes ▼	

Fig. 22.7 Print screen of Google-table "Table of a student"

The described technology has the advantage that the programs attached to tables are available to the teacher, which can them modify if necessary. With the knowledge of the programming language of JavaScript, which the teacher leading IT discipline, it is easy to handle the programs. Similar tables can be used for other activities.

22.3.3 Monitoring Activities

When conducting an exam or a test in writing, tickets (conditions of tasks) are traditionally used. Instead of paper tickets/tasks, it is convenient to use electronic documents that are distributed to accessible folders in students' computers or, as in this case, in folders on a Google disk. Random distribution, of course, can be done automatically. To do this, a special table is organized, which is generated when creating the group folders (Fig. 22.8). In the first column, there are the links to the student folders, in the second one, there are the first sequential numbers are written. Using the standard function of Google table, the numbers are mixed, after which the function "Distribute tickets" is launched through the menu. The ticket texts must be preplaced in a folder that is referenced in cell ExamCards-WP. As a result, the tickets are copied to the student folders according to the allocation of numbers. At the same time, documents are created for recording ticket responses. These documents automatically record the student's name simultaneously with the conditions of the tasks (wording of questions). At the end of the exam, the teacher starts the End of Exam function, and the access status of documents with the answers changes from Editor to Reader.

The same technology can be used for the intermediate evaluations. If the responses require multiple files to be created, subfolders are generated in the student folders, and links to them are also placed in the third column.

The ticket generation is also easy to automate (Fig. 22.9). We suppose the ticket includes two questions that can be repeated on different tickets. An examination ticket template is created, in which the parts to be replaced are marked:

PI3-1	ExamCards-WP	Answers
Alekseev	card-14	anwer-14
Abramov	card-15	anwer-15

Fig. 22.8 Print screen of Google-table "Table of examination"

Fig. 22.9 Exam ticket template

Examination ticket No.
Question_1
Question_2

Possible texts of questions are combined into the documents “Questions_1” and “Questions_2”. After that, the generation script creates the files “Ticket_1”, “Ticket_2”, etc., in a separate folder, setting the number and replacing the texts “Question_I” with one of the questions randomly or sequentially.

22.3.4 Coursework and Projects

For a course work, the same actions as for laboratory works are necessary: the formation and distribution of topics, performance control, and evaluation. The peculiarity is that, as a rule, the conduct is regulated by documents of the level of the department or dean, which determine the attachment to the teacher, the formation of topics, the stages of implementation and the delivery deadlines. In this case, the teacher is required to tighter control of the progress. For such a work, the same table is suitable as for the group, but instead of tasks there will be stages of work. A shared folder is created for the group, in which the activity log is maintained with the assignment of responsibilities and milestones.

22.3.5 Diploma

In many ways, the guidance of the diploma work is similar to the guidance of the semester work. Hereby, the difference is in more work and stricter management control. In some universities, such stages as the distribution of topics, the monitoring of implementation, the organization of checks for anti-plagiarism and others are automated. The teacher interacts with the student for a longer time, so fixing the correspondence, discussions, and comments in one place using the same Google disk is very appropriate. A communication online is simply necessary, since a student during the performance of diploma rarely appears at the university and, therefore, does not meet with the supervisor of diploma work.

It is advisable to keep a table of your graduates, noting the work progress and conducting the correspondence through Google documents. As for the group of students, the table shows the date of change of the document “Correspondence”. There are the links to the folders with diploma materials. In a separate table, the teacher can capture the documents that are available to all the participants at once using a special function.

22.3.6 Communication Means

E-mail, with which teachers usually communicate with students, is not always convenient. Sometimes you have to search for previous messages for a long time or send the

same messages to different students. To keep the correspondence in one document, we will use the option to work with the group (Fig. 22.1). By selecting a separate column with check-box elements, we send a message to those students who will have a check mark on the check box. Messages are sent by mail and simultaneously duplicated in the document "Correspondence" while the students write answers directly into the document and do not clog the teacher's mail. If mail communication has occurred, one can use a special script to transfer messages to this document.

The management has a similar purpose to simplify the communication. The manager must sometimes notify everyone about something while receiving the notifications, or make a survey. For these cases, Google forms may be useful. On the instructions of one of the managers, a module was created to simplify the communications. The module can work either autonomously or as a part of the Department system. When a user logs in, he/she has Send Messages button and an information about the number of new incoming messages. When the user enters the button, the list of teachers is displayed. Selecting the ones to which the message is sent, the user writes the text and notes the message type: "No"—without notification, "Ok"—with notification, "Yes/No"—the corresponding answer is required, and "Text"—the usual answer. The user can mark the message as duplicated in a regular mail.

The recipients display the date and time of the message, the sender's last name, and the text of the message. The content of the response cell depends on the message type. When answering yes/no type, two clicks are sufficient, one to a message with notification.

The sender sees the answers in a convenient form: the lists of respondents are displayed in the affirmative or negative. Similarly, after a little refinement, the possibility of voting or answers to the questionnaire can be realized.

22.4 Software Tools for the Individual Use

One can free up the teacher time in a simpler and more accessible way. The fact is that a routine work, devouring time, largely concerns the teacher himself, without affecting his/her communication with the students and colleagues. This includes the mentioned preparation of educational materials, attendance, and student performance, and the process control (performance of independent and laboratory works, course and diploma projects). This also includes the work with graduate students. It is clear that the teacher does not work in an isolation, but in the information space. However, the time for exchanging information with the external environment is significantly less than the time for his/her independent work. Of course, we exclude the case of conducting training sessions. We consider some very simple approaches.

Hereinafter, a short description of such software modules as "Attendance", "Control Actions", "Educational Process", and "Preparation and Conduct of the BD-exam" are given in Sects. 22.4.1–22.4.4, respectively.

№	Seminar	3	3	3	3	3	3	3	3	3	2	0	33	pair
	Name	23.9	30.9	7.10	14.10	21.10	28.10	4.11	11.11	18.11	25.11	2.12	abs	%
1	Gurinov Aleksander					abs	ill						4	12%
2	Krivenko Dmitri		abs	abs	abs		abs	abs	abs				15	45%
3	Levshtanov Denis	abs	abs	abs	abs	abs	abs	abs					15	45%
4	Samsonov Gleb			abs									3	9%
5	Semchenko Maxim												0	0%

Fig. 22.10 Attendance of classes

22.4.1 Attendance

In the traditional administrative environment, paper attendance logs are still widely used. Usually they are at the head of the group. At the end of classes, the teacher reviews the magazine and signs in it. If only the dean needs it, for the teacher it is nothing more than a meaningless ritual. But if he does not care about the attendance (this is a significant feedback), he/her should keep his/her records too. A spreadsheet in Excel is suitable for this. Figure 22.10 shows the view of such a table. Everything is clearly shown here: the list of students (formed at most once per semester), the dates of classes (recorded from the schedule), and the presence on pairs. Attendance by the student and the day are automatically considered, for clarity, the percentage of attendance is painted and a graph is drawn depending on the attendance. Five minutes of time spent during classes more than compensates for the ability to control the process.

22.4.2 Control Actions

Consider the situation at the semester end. By this time, the necessary laboratory work must be completed and evaluated, the accounting of which is also carried out in similar tables (Fig. 22.11).

Note that the list is not re-entered, it is taken from the previous sheet automatically. The "var" column is the option number. The same number means that the task is command (the manager is highlighted). The result is the average grade; it is a part of the final grade for the exam. This subject does not provide a task, only two questions in the ticket. The color of the cells marked the grades that the teacher is ready for students

№	Group 106M - 1-2019	Practical work					Exam		Mark	%abs	Mark+abs
	Name	var.	IDEF0	ERD	DFD	Total	№ 1	№ 2			
1	Gurinov Aleksander	8	5	5	5	5.0	4	4	4.3	12%	5.4
2	Krivenko Dmitri	8	4	4	2	3.3				45%	
3	Levshtanov Denis	11	3	4	2	3.0	4	5	4.0	45%	3.1
4	Samsonov Gleb	8	4	5	5	4.7	3	3	3.6	9%	4.6
5	Semchenko Maxim	1	4	5	3	4.0	3	5	4.0	0%	5.7

Fig. 22.11 Personal examination list of the group

Data Bases-2020

№	Name	Task		Ticket			Total	Note		7.9	21.9	19.10	2.11	9.11	Lab.
		№	Mark	№	Mark	Mark									
	08-302														
1	Bazylnikova E.	7	5	23	3	2	**3**		1	+					5
2	Gribtsova A.	24	5	14	5	5	**5**		5	+	+	+	+	+	5
3	Oratovsky E.						**0**		2	+	+				
4	Zhelyabin I.	11	5	8	5	5	**5**		4	+	+	+		+	5

Fig. 22.12 Personal examination list of the course

to set automatically, taking into account their work in the semester. The resulting final grade is adjusted depending on the student's attendance, which is consistent with the recommendations of The Ministry of Science and Higher Education. Of course, a rounded grade is placed on the list. Many years of experience have shown the fairness of this approach both on the part of the teacher and on the part of students.

A fragment of this list, for several groups, is shown in Fig. 22.12. During an in-line lecture, it is unrealistic to conduct some surveys; students on the control day simply note their presence on the leaflets. And, surprisingly, better students show on average the best results in the exam.

22.4.3 Educational Process

IT training requires a laboratory work. In addition to the quality control, it is necessary to take care of the timely preparation and delivery of the work results. This equalizes the semester load of both students and teachers and helps improve the result. In addition, by developing a habit of performing the work on time, the student facilitates his/her life in a real production environment in the future.

Figure 22.13 shows the table of records of laboratory works performed by student teams.

For each work, two columns are allocated: a group estimate and a check-sum, which is formed from individual assessments of participants (0 is the sum coincides with the estimate). In columns marked as checkpoints, there are real deadlines for delivery. Missing the term results in a score loss. The absence of delivery marks means that the team violated the last term.

A similar, but in a sense more stringent control is carried out during a thesis work (Fig. 22.14). Of course, no points are put here; just the participants see their place in

Magistracy, 1 - 2019-2020

Command's work

№	*Name of works*	*Leader*	*mmb*	*IDEF0*		*DFD*		*Use-Case*		*Class*		*Activ*		*Aver.*	Control points 17.10	31.10	14.11	28.11	12.12
1	Elaboration program project		0				0		0		0		0	0.0					
2	Library of Univercity	Sitnikova Olga	0	12	12	11	11	12	12	14	14	10	10	11.8	10.10	31.10	7.11	21.11	28.11
3	Hotel	Stesnyagin Semen	3	14	2	9	–2	14	2	11	–3	12	2	12.0	17.10	31.10	7.11	21.11	21.11
4	Elaboration of program project	Stanovskaya Yana	3	14	0	10	1	15	1	15	4	14	2	13.6	10.10	17.10	14.11	14.11	14.11

Fig. 22.13 Performance of laboratory works

Control of graduates 2020

№	Name	14.2	21.2	28.2	6.3	13.3	20.3	27.3	3.4	10.4	17.4	24.4	1.5	8.5	15.5	22.5	29.5
1	Bashkatov Victor		art								dip		ans	rep	dip		
2	Goobin Maxim	rep	art		dip	ans		dip	dip		dip	dip		dip			
3	Zabelin Mikhail	art		rep		dip			dip								
4	Kolyandra Dmitry	dip	dip	dip	dip	dip	dip			dip	dip	dip					

Fig. 22.14 Performance of thesis works

the group of graduates and adjust their activities. Such a picture really allows you to reduce overload before protecting work and gives a chance to a better preparation.

Of course, all these materials are open, but students cannot change them. Therefore, the original remains with the teacher, and the copies are sent to the participants: in the case of a diploma—to students, in the rest—to the headmen of the groups.

22.4.4 Preparation and Conduct of the BD Exam

Mandatory part of the database exam is tasks' solving. The simplest, but undoubtedly, a useful variant of the task is a SQL query to a given database. The stage of preparation of the task (this is the teacher's case) and the procedure for passing the exam (the main participant is a student) are considered.

In preparation, the teacher identifies and describes the subject area, designs the database structure, then formulates the request and writes its solution to the system database (Fig. 22.15).

Note that no query condition is generated here. Automatically adding elementary variations (e.g., attribute names and values) is not difficult, but reduces the write-off protection.

The prepared examination material is offered to the students in the exam. According to the exam ticket number, the student receives a problem to solve. The difficulty of evaluation is that the right solution is not always the only one. Therefore, the validation criterion is that the query results match. With low probability, the correct result can be obtained when the query is incorrect. However, a deliberately incorrect request, of course, does not pass (Fig. 22.16). The correct answer is given in Fig. 22.17.

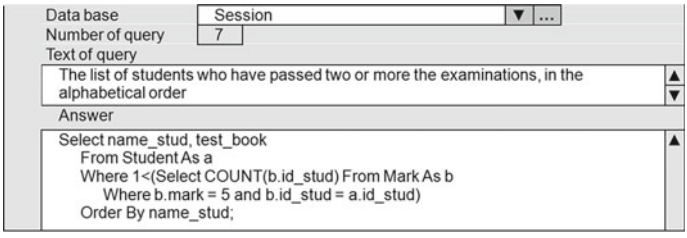


Fig. 22.15 Feedback control

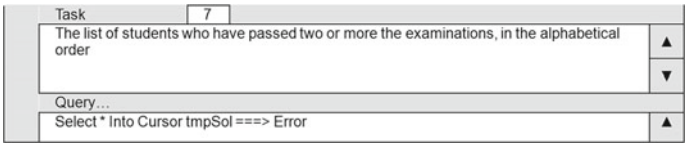


Fig. 22.16 Error in query

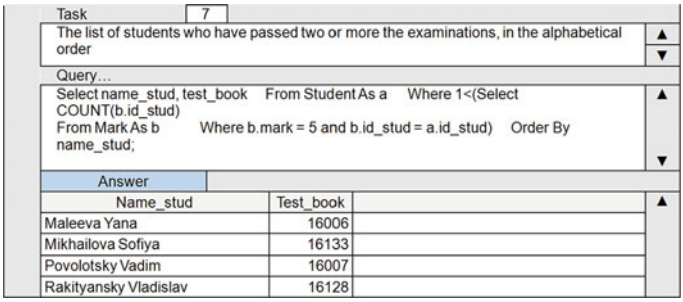


Fig. 22.17 Correct answer

Thus, despite the fact that the proposed tools are quite simple, not related to the data exchange on the global network, they occupy their niche in the work of IT teacher and allow him to liberate a certain amount of time not only without losing the quality, but also with its improvement (monitoring the learning process, preparing and implementing the task options).

22.5 Conclusions

As the experience of the authors shows, the use of the described software significantly facilitates the learning process, making it more orderly and effective. When all the teachers of the same department use the same software, this will facilitate the work and leadership of the department. Reducing the labor on routine operations will allow teachers to focus on methodological work. It is necessary to provide a communication with external software systems, portals of universities. There are no such problems on the part of the described means. However, not always the university system has means of a convenient program access, which by itself slows down integration.

References

1. Abramov, S.M.: Mistakes in state supervision of higher education are main problem of higher education in Russia. In: 13th Conference on Free Software in Higher Education. Conference

materials. Pereyaslyavl, M.: Basealt, pp. 34–38 (2018) (in Russian)
2. Pavel, A.-P., Fruth, A., Neacsu, M.-N.: ICT and e-learning—catalysts for innovation and quality in higher education. In: 2nd Global Conference on Business, Economics, Management and Tourism, Prague, Czech Republic. Procedia Economics and Finance 23, pp. 704–711 (2015)
3. Finley, A.: A Comprehensive Approach to Assessment of High-Impact Practices. 2019. https://secure.aacu.org/iMIS/ItemDetail?iProductCode=E-HIPSASSESS. Last accessed 2020/07/21
4. Isaeva, I.E., Churikov, M.P., Kotlyarenko, Yu.Yu.: Effectiveness of evaluation of the activities of university professors: comparison of domestic and foreign methods. Internet J. Sci. Stud. **7**(3) (2015) (in Russian)
5. Modular Object-Oriented Dynamic Learning Environment. https://moodle.org/?lang=en. Last accessed 2020/07/21
6. Easy LMS. https://www.easy-lms.com. Last accessed 2020/07/21
7. WizIQ Virtual Classroom. https://www.wiziq.com. last accessed 2020/07/21
8. NextThought. https://www.capterra.com/p/154996/NextThought/. Last accessed 2020/08/04
9. Populi. https://sourceforge.net/software/product/Populi/. Last accessed 2020/07/21
10. 1C-University. https://solutions.1c.ru/catalog/university-prof. Last accessed 2020/07/21 (in Russian)
11. OpenEduCat. https://sourceforge.net/software/product/OpenEduCat. Last accessed 2020/07/21
12. Collegix. https://sourceforge.net/software/product/Collegix. Last accessed 2020/07/21
13. Best LMS Software 2020 Review. https://www.capterra.com/learning-management-system-software/. Last accessed
14. John, S.P.: The integration of information technology in higher education: a study of faculty's attitude towards IT adoption in the teaching process. Contaduría Y Adm. **60**(1), 230–252 (2015)
15. Samochadin, A., Raychuk, D., Nosnitsyn, S., Khmelkov, I.: A comprehensive set of mobile services supporting education process. In: 4th World Conference on Educational Technology Researches, Procedia Social and Behavioral Sciences 182, pp. 613–618 (2015)
16. Lukin, V.N., Chernyshov, L.N.: Technology of student knowledge control in the discipline of the database. In: Scientific Works of the XX Anniversary All-Russian Scientific Conference on Engineering of Enterprises and Knowledge Management. Moscow, FSBOU VO REU named after G.V. Plekhanova, pp. 150–155 (2017) (in Russian)
17. Source codes Google-scripts. https://github.com/LevChern/eduprocess. Last accessed 2020/08/04

Chapter 23
Entity-Event Ontology Construction by Conceptualization of Mentions in Text Corpus

Michael C. Ridley

Abstract Knowledge-based systems constitute a powerful tool for tackling and navigating complex domains, but they have the potential to be employed more often in practical tasks if some obstacles are cleared. Creating and keeping knowledge bases up-to-date is a challenging problem without automatic extraction of knowledge from data sources like documents. One of the solutions is ontology learning, which enables automatic construction and population of ontologies used to store knowledge. This chapter proposes an automatic method for domain ontology construction based on extracting entities and events from texts. Also, it is stated that upper-level template ontologies used when analyzing text corpus are suitable for creating target instance ontologies that describe a specific domain. The task of instance ontology construction is formulated in the terms of reconstructing real-world events via analyzing their mentions in a text corpus and structuring them according to the template ontology. This method allows an automatic analysis of big volumes of textual data like posts from social networks, news, contracts, specifications, etc., by utilizing natural language understanding tools used to extract domain knowledge. We developed a system that collects texts from the Internet, analyzes them, builds an ontology, and presents it as a knowledge base. One of the current applications is in optimizing business processes in a domain of civil aviation: document management, sorting and navigating documents, text summarization, semantic enterprise search, and exploratory search. Furthermore, it is claimed that extracted knowledge can be used to construct informative features in machine learning tasks.

23.1 Introduction

One of the dominant applications of information systems is the analysis of numerical, categorical, and other structured data. As an example, CRM systems often operate on

M. C. Ridley (✉)
Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, Moscow 125993, Russian Federation
e-mail: mr@kalabi.ru

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_23

manually entered or imported data about relations with customers and sales scripts. Another classic example is a system for collecting sensor readings and storing them in a transaction processing database along with OLAP cubes for analysts to prepare reports for managers.

Even so, practical tasks often require direct application of entities, relations, entity attributes, and rules that govern them. It is required in geospatial services, question answering, master data/record management systems, document management systems, enterprise search, content management systems, and so on.

Online media monitoring is a classic case in different field of study. Since the Internet is a kind of mirror for the real world, it contains the implicit mentions of real-world events. Many related problems such as reconstitution and prediction of events, explorative search, and early trend detection can successfully be formulated in terms of ontology construction or ontology learning. Such ontology is a useful source for computing domain-specific metrics like event importance, surges in topic discussions, vocabulary diversity when discussing something, indirect links between entities, semantical similarity of mentions in different languages.

In the present context, it is common to bring up the concept of a semantic Web as a way of enriching data with meaning by means of a special markup and communicating metadata and knowledge among Web resources. If it was widely adopted, semantic markup of Web pages would render information on the Internet machine-readable. In a sense, it helps to establish a connection between text and its conceived meaning. One of the things that semantic Web brought to us is Web ontology language (OWL), which is capable of representing classes, individuals, relations, and even reasoning on top of described ontologies [1].

Similar ideas are found in the enterprise software domain. For example, an ordinary full-text search could be superseded by a semantic search capable of navigation, filtering, grouping content with the same meaning, linking, and so on. Theoretical foundation of such cases is extraction and representation of knowledge by means of ontologies. These ontologies typically consist of entities or concepts, attributes, relations, and slightly less common—axioms and rules.

Unfortunately, most ontologies are still constructed and even populated manually. This approach is expensive and time-consuming. It cannot be used at all in case of large data inflow or when data has to be processed in a real-time fashion. Also, manually constructed ontologies tend to be too general and thus form a habit of underestimating the power of applying ontologies to real-life tasks. It is also interesting that early automatic ontology construction projects focused on generic ontologies as well: usually, it was about extracting hyponyms/hypernyms and meronyms. Extraction of non-taxonomic relations was not common for some time.

Automatic ontology construction is an important problem that requires extracting the entities, concepts, and relations between occurring in a text corpus. For example, analysis of corporate documents and e-mails leads to maintaining a knowledge base with a map of all operations, document templates, workflows, etc. Analysis of mentions in the Internet enables to build the predictive and descriptive models of real-life events and their dynamics. By analyzing posts and chats of a person, one can build the contextual profiles: what person talks about and how, what he or she likes,

what are links between topics according to this particular person, and so on. Using such profiles, broad topics like "Politics" can be broken down into attitudes toward specific politicians. Also, they can be employed to improve the personalization of targeted ads.

Needless to say, same methods can be utilized in a broad spectrum of information extraction tasks. For instance, it can be used to create domain-oriented search engines and explorative search systems.

When information is extracted from documents according to some meta-ontology, resulting database can then be used to provide rich navigation features, explicit and implicit link analysis, grouping mentions by things they refer to, structured semantic queries, temporal analysis, single-document and multi-document summarization, and other tasks. The same can be said about some natural language understanding tasks like question answering and textual entailment.

Our contributions are in the proposal of template-level and instance-level ontologies, introduction of entity-event ontologies, development of the method for construction entity-event ontologies from text corpora, implementing a full-featured system based on the method and capable of analyzing mentions on the Internet, conducting experiments for quality evaluation, and deployment of the system in civil aviation organizations.

The chapter is organized as follows. Section 23.2 provides an overview of ontology learning methods. Section 23.3 introduces a novel method of entity-event ontology construction along with practical and theoretic considerations. Section 23.4 describes system implementation and aspects of collecting data from the Internet. Section 23.5 concludes the chapter.

23.2 Related Work

Ontology construction, enrichment, and population attempts are all related to ontology learning—the act of acquisition of a domain model from data [1]. Input data can be in any form ranging from structured XML documents to semi-structured HTML pages and unstructured raw natural language text. In the latter case, it is ontology learning from text [2]. The task then is extracting conceptual knowledge from text input and building or populating an ontology from it.

It is useful to note that ontologies can be very different: some of them are general, some of them are domain-specific. Also, often they are not full-featured: for example, restricting the original problem to a taxonomy case is rather popular. There is a concept of a semantic spectrum that allows describing knowledge representations in terms of expressiveness ranging from glossaries (simple lists of terms) to controlled vocabularies, data dictionaries and thesauri, data models, taxonomies, and finally full-features ontologies.

Ontology learning is a broad field of study that has different classifications [3, 4]. Usually, researchers divide methods based on the following characteristics:

- Type of input data:
  - Structured: DBpedia, relational databases, some XML.
  - Unstructured: arbitrary text in one of natural of artificial languages.
  - Semi-structured: unstructured data with structured parts such as Wikipedia articles based on templates, or financial statements.
- Level of automation:
  - Semi-automatic: requires user intervention.
  - Automatic: after the model is ready, no control is needed.
- Learning targets:
  - Concepts and instances.
  - Relations: taxonomic and non-taxonomic like thematic roles and syntactic relations.
  - Axioms: used to model sentences that are true and to create new knowledge from the existing one.
  - Meta-knowledge: rules of how to learn ontology, what attributes can be extracted.
- Purpose of the method:
  - Creation of ontology from scratch.
  - Ontology population and updating.
- Learning techniques:
  - Linguistic: syntactic analysis, morpho-syntactic analysis, lexico-syntactic pattern-parsing, terminology networks, syntactic frames, and text understanding techniques.
  - Pattern/Template matching[1]: Hearst patterns, regular expressions, exception templates, and symbolic interpretation rules.
  - Logical: inductive logic programming, clustering, and rule learning based on first-order logic or propositional learning.
  - Statistical: hidden Markov models, sequence models, neural networks, conditional random fields, co-occurrence data, bag-of-words, and so on.
  - Combined/Hybrid: various heuristics using statistical methods on top of linguistic features. Applying methods depending on the context like WebKB uses first-order logic rule learning along with Bayesian learning.

As this field of study is active, there are lots of tools like pioneer Text-to-Onto, OntoLT, WebKB, DODDLE II, CRCTOL, C-Pankow, Sofie, and others based on a

[1]It is worth noting that pattern matching is a common choice in information extraction tasks as it provides very high precision, although by the cost of lower recall and constant maintenance of rules for every supported language.

broad range of learning methods like formal concept analysis, clustering, association rules, linguistic patterns, regular expressions, statistical methods, graph theory, probabilistic graphical models, and so on [4].

Among widely acknowledged problems like axiom learning, there is also ongoing research in the field of language-independent methods that are maintenance-friendly and capable of learning structure on their own. One of the most promising methods is based on neural networks and treating expressive ontology learning as a machine translation task and mapping sentences to axioms [5].

23.3 Construction of Entity-Event Ontologies from Texts

Ontology learning is one of the methods for extracting conceptual knowledge from texts. Those ontologies are used to partly represent the meaning of the text or at least its structure. Often there are predefined classes in ontology, and the task is to extract their instances by using pattern matching or other extraction methods. Another case is when there is fixed base ontology and the task is to construct a domain ontology by the means of extracting relations, classes, their instances, and so on. The term "base ontology" is used here to denote that it can be both upper-level ontology with a specific focus or just a sufficiently high-level domain ontology.

As mentioned earlier, most automatically constructed ontologies are focused on universal linguistical concepts and usually employ abstract relations like IS-A, PART-OF, INSTANCE-OF, HYPONYM, HAS-VALUE, and others. Although suitable for artificial intelligence research, it has low practical value in applied knowledge storage and representation tasks. Applied knowledge bases and knowledge-based systems typically utilize domain ontologies as they benefit from their focus on specifics and predefined relevant formalism.

However, there is a space in between the following: some problems require capturing of new ontological knowledge in a broad, but highly structured domain. This kind of problems can be tackled by using two levels of ontologies—template and instance ones—as outlined below:

- Template ontology acts as a description of what can happen in a world from some practical viewpoint and sets a general structure for knowledge in supported domains.
- Instance ontology is formed by extracting information from a set of documents according to template ontology and de-facto constitutes a knowledge base.

Template or base ontology can be just an upper ontology like DOLCE, SUMO, Cyc, and others, but that is not always the case, as it can be domain ontology as well. It is quite common for applied knowledge bases and some cases of expert systems.

For example, one could define template ontology for representing knowledge in court orders and then build a court order knowledge base in form of ontology. Another case is a knowledge base of the company's legal documents that can be

used by employees to navigate through complex procedures, sets of legal templates, contract provisions, and so on.

Speaking of news, politics, general or corporate events, legal announcements, and social media posts, on some level one can say that they share a lot of commonalities. One way of capturing these commonalities is a method of entity-event ontologies.

The method of entity-event ontologies is based on the idea that as something important happens in the real world, it leaves traces in multiple mentions in documents, news, social network posts, and so on. Real-world events are scattered across numerous mentions of them. All those mentions combined describe a single real-life event—for example, company acquisition, product announcement, legal actions, protests and attacks, or natural disasters.

An example of suitable ontology for describing real-life events is rich event ontology [6]. Unlike general ontologies, it does not underrepresent events and treats them explicitly by introducing concepts like participants, causal and temporal relations, special relations like HAS-PRECONDITION, HAS-RESULT, HAS-SUBEVENT, and so on.

Similar ideas are also implemented in Thomson Reuters (Refinitiv) knowledge graph for financial and capital markets analysis. It models many real-life event and entity types like organizations, quotes, regulators, assets, supply chains, deals, industries, exchanges, company officers, and so on.

In our system, we used entity-event ontology that is based on real-life events and entities, their mentions (instances) in texts, and context information derived from the text and other sources. Each real-life event instance has some attributes depending on its type. For example, natural disaster events have attributes disaster type, location, time, deaths, other casualties, related entities, and mentions.

During pilot operation of our system, we have identified features not universally present in similar systems, but useful and important for solving practical knowledge base-related tasks:

- Spatio-temporal analysis can be utilized for creating chronologies of how events unfold and how they were discussed. Also, it helps to identify and track planned events like protests. Both time and place can be specified on their own or be relative to other temporal or spatial statements.
- Time and geographical "intervals" and hierarchy. When considering all elections in Russia, it is important to consider mention in all of its cities and regions. Also, time intervals can have different precision ranging from seconds to years and centuries. Temporal and spatial hierarchies enhance quality of merging different mentions of presumably same real-life events when they have different descriptions.
- Language-independent ontology. Although text processing methods in the system are language-dependent, the resulting facts are independent, so that one real-life event can have multiple mentions in different languages.
- Special events for indirect speech and quotations are handy for analysts as they get an option to compare different judgments on the same topic, validate expert opinions, and search for discrepancies in someone's opinion. As they also have

time and place, it is possible to analyze statements like "John said yesterday that Sandy is going to be in Portland tomorrow". If opinion was published July 4, then first event is "Sandy visiting Portland on July 4", and the second one is quotation event published on July 4 and containing John's opinion as of July 3 and a reference to the first event.

- Temporal analysis requires culture detection because depending on language, hemisphere, and other contextual data, it is difficult to interpret dates, time of year, the first day of a week, and so on [7].
- When working with constantly changing data, it is important to preserve fragments of text with mentions of event. For example, mainstream media sources often edit news articles—and being able to see this is useful for analysts.
- Sentiment analysis is not related to ontologies, but maintaining author sentiment for every mention is convenient for analysts. For example, it enables analysis of general sentiment regarding authors from specific countries or specific types of media.
- Sometimes there is no rule or event type available for linking adjacent entities in text, but for practical reasons, this kind of links should also be preserved and presented to users. It can be done with a special attribute "related entities" in every event.
- Ontologies and knowledge bases are often viewed as something static, but as they have the power to be the ultimate source of data in knowledge-based systems, it is wise to continuously update them as new data arrive and populate them with new facts. For instance, ontologies can easily be a data source for an interactive dashboard or search system.
- In practice, it is often better to implement knowledge bases on top of conventional technologies like document-oriented or relational databases and encapsulate them through API. For some reason, software development engineers are sometimes biased toward technologies like RDF, OWL, SPARQL, logic programming, Prolog, first-order logic, and try to avoid them. As these technologies are seldom mainstream, they obstruct widespread adoption of ontologies. Thus, researchers need to communicate that knowledge-based systems are not tied to research community tools and can make use of conventional technologies as well.

23.4 Implementing Ontology Construction for the Internet

Internet and other sources nowadays are acting as a mirror of real-world events: Every second, countless people send and describe things that they experience, companies upload and produce tons of documents, armies of journalists and bloggers interpret and follow events, conduct citizen investigations, and analyze different sources. For example, every minute, Twitter users send more than 511,000 tweets, Tumblr users publish 92,000 posts, 188,000,000 million emails are sent, and 277,000 Instagram stories are posted [8].

In some sense, the Internet along with social networks can be considered a giant crowdsourcing platform for a wide variety of topics. It is well-known that general-purpose event information such as published in mainstream media can be extracted from the Internet as well. However, it is also true for a wide range of specialized topics like cybersecurity. For instance, 75.8% of CVE vulnerabilities related to the Linux kernel were exposed before their official disclosure as 0-day vulnerabilities with the average time advance of 19 days [9]. Also, 100% of NIST CVEs were also published and described on Twitter [9]. Another example is disaster and emergency monitoring used by agencies such as US FEMA and the UN Office for Coordination of Humanitarian Affairs for day-to-day operations [10].

It must be noted that along with the aforementioned abundance of relevant data, the Internet has some special traits:

- It has to be scanned regularly and in a distributed fault-tolerant manner as new information is generated extremely fast and is very diverse.
- Storing all collected data is economically unfeasible and technically impractical, thus requiring special tactics for getting rid of unneeded data.
- Relevant niche sources like specialized groups in social networks are important for achieving minimal delays.
- As multiple sources and people discuss same events, their descriptions are diverse, written in different languages, duplicated, and merged in other discussions. Also, people might have inconsistent views on the event. Even more important, most significant events evolve and change over time.
- All sources have different markup, styles, page organization, and so on. Robots have to be adapted for all major sources and be smart enough to extract text with decent quality for secondary sources. Also, some sources like social networks provide API directly or via data provider services like GNIP.
- Extracted text usually contains banners, ads, and other disturbing content that has to be removed.
- Each extracted text appears in some context: media type, source URL, publication date and time, and for social media—authors, likes, reposts, etc. This context is very helpful for both end-users like professional analysts and for the system itself. It allows analyzing bias and sentiment of sources and authors toward different topics, country and language biases, topics with very little interest in social networks but forced by mainstream media, negative sentiment from people along with a positive sentiment from mainstream media, silence on some topics by official media, and other notable cases.

As Fig. 23.1 illustrates, texts on different natural languages should be analyzed via a set of NLP tools and analytical modules to extract entities and events and populate ontology with them. Also, ontology is recurrently updated with information from external sources like GeoNames which provides geographical hierarchy, DBpedia, and WikiData. External ontologies capture useful information such as government officials, companies and their structure, industries, and technologies.

The system uses various NLP tools like Tomita parser, StanfordNLP, OpenNLP, and Rosette EX along with custom extractors based on regular expressions, pattern

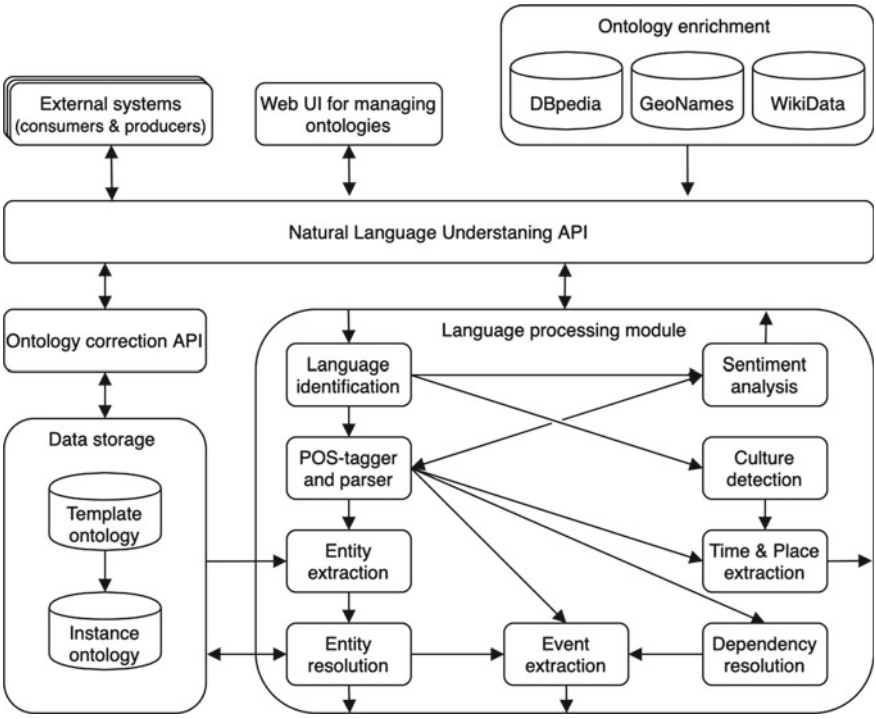


Fig. 23.1 High-level architecture of computational linguistics subsystem

matching, and conditional random fields. The latter method is superior to other ones like maximum-entropy Markov models because it has no label bias and does not require modeling dependencies among observables. In addition to NLP tools, analytical modules are used: culture detection, ontological part of entity resolution, and so on.

Eventually, all events are labeled with type, temporal information, entities involved, and which event attribute they belong to, related entities that are not involved, mentions and their sentiment, source, and external context.

Our system is organized in layers as it is shown in Fig. 23.2. Each layer consists of a set of microservices communicating through a messaging system. Web harvester subsystem is responsible for mining text data from the Internet. It takes a list of sources and robots as input and produces unstructured texts free of unwanted content and context: time of access, time of publication according to the source, author, URL, etc.

Computational linguistics subsystem takes as purified texts from the harvester subsystem as input. Its output is a set of XML-formatted text fragments with highlighted entities, events, temporal and spatial labels, and so forth. It should be emphasized that this subsystem is the only place in the system that depends on a language. Support of a new language is implemented by adding a new module for it in this

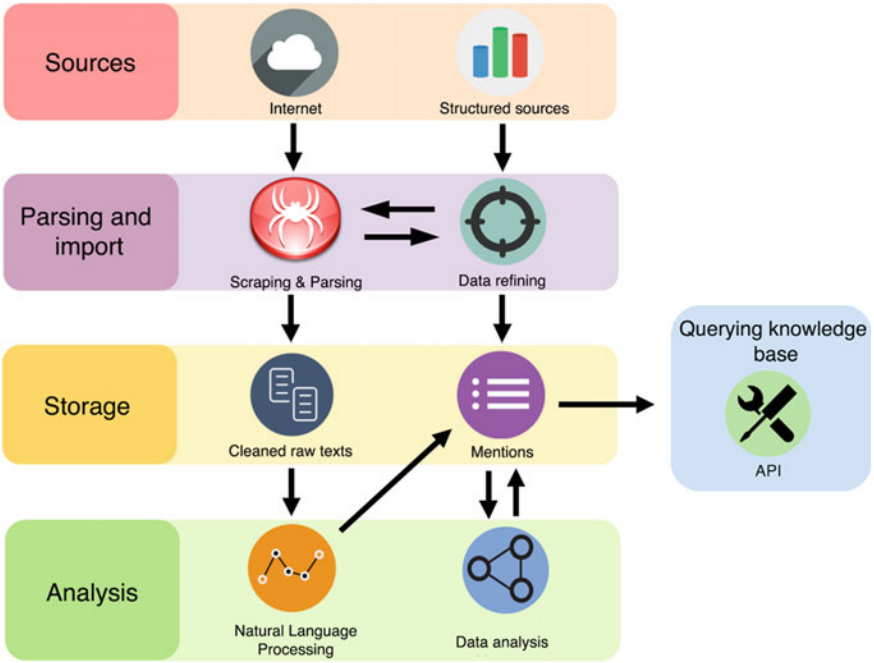


Fig. 23.2 Dataflow between system layers and modules

subsystem. Everything else in the system is dependent on extracted facts, but not the text itself.

Storage subsystem takes from XML-formatted snippets as input and parses them. As a result, mentions are formed and stored across a distributed NoSQL database and full-text search solution. Also, integration module of the subsystem provides an external REST API for querying "ontology slices" in JSON by using structured queries formed as a set of events, entities, and attribute filters. This API is used by Web UI of a system as well as external client systems.

Data analysis modules recurrently visit mentions storage for a purpose of refining and confirming data in the ontology and supplementary data. Some of them are data enrichment modules: they weaken unlikely events, compute domain-specific metrics for users, merges event mentions into real-life events, etc.

23.5 Discussions and Conclusions

We developed a system that collects massive amounts of texts from the Internet, analyzes them, builds the entity-event ontology, and presents it to the end-user as a knowledge base [11, 12].

As it is always hard to evaluate ontology learning and relating tasks, we used a slightly different evaluation approach. Given that system supports structured queries and can be used for exploratory search, we formulated 15 queries with interpretations like "Which countries suffered from earthquakes and floods in 2015–2019?", "What are Vladimir Putin's international visits in summer 2019?", "What exploit kits are made in 2018?", "What world leaders did Donald Trump meet in years 2018–2019?". We used human assessors to obtain true answers for each query.

Comparing true answers with the system's ones, we found that precision is 0.91 and recall is 0.68. It should be noted that precision is substantially high, while recall is unremarkable. One of the reasons is that when some event is described using a set of rules, they are usually relevant and rigorous, but they are always incomplete: it is difficult to capture all ways of saying something. This problem can probably be solved by taking advantage of new methods like neural machine translation mentioned in [5]. Although it is unlikely that they are capable of achieving such high precision nowadays, they can possibly be combined with the existing ones to improve recall without significant loss of precision.

The first release of a system was tailored to the needs of civil aviation and was used in the following tasks:

- Safety occurrence reports' analysis in Aviation Safety Network.
- Structuring and navigating through technical specifications and local regulations.
- Social media interaction and brand monitoring for airlines.
- Monitoring and analysis of international civil aviation news and new regulations.
- Early risk identification (natural disasters, political changes, etc.).
- Exploratory enterprise search among contracts, documents, supporting documentation, work regulations, instructions, reports, etc.

Another result is that extracted information can be used as informative features in machine learning tasks for the target domain. For example, we used system to predict protests in Moscow, Russia during 2011–2020 based on social network data with an overall accuracy of 72.4%. Also, it was successfully used to enhance document classification and clustering algorithms by providing high-informative extra features.

Moreover, it can be viewed as an automatic text corpus processing method that allows using of classic statistical and data analysis methods by extracting domain-specific information from text. As extracted knowledge is highly structured and easily operated, it can be used by such methods without any further reference to the source texts.

References

1. Maedche, A., Staab, S.: Ontology learning for the semantic web. IEEE Intell. Syst. **16**(2), 72–79 (2001)
2. Buitelaar, P., Cimiano, P., Magnini, B.: Ontology learning from text: methods, evaluation and applications. Frontiers Artif. Intell. Appl. **123** (2005)

3. Al-Arfaj, A., Al-Salman, A.: Ontology construction from text: challenges and trends. Int. J. Artif. Intell. Expert Syst. **6**(2), 15–26 (2015)
4. Shamsfard, M., Barforoush, A.: The state of the art in ontology learning: a framework for comparison. Knowl. Eng. Rev. **18**(4), 293–316 (2003)
5. Petrucci, G., Rospocher, M., Ghidini, C.: Expressive ontology learning as neural machine translation. J. Web Semantics **52**, 66–82 (2018)
6. Brown, S., Bonial, C., Obrst, L., Palmer, M.: The rich event ontology. In: Proceedings of the Events and Stories in the News Workshop, pp. 87–97 (2017)
7. Ridley, M.: Software tool for extraction of temporal context from text documents. In: Proceedings of the XI International Conference on Nonequilibrium Processes in Nozzles and Jets, pp. 577–578 (2016) (in Russian)
8. DOMO Data Never Sleeps 7.0 Infographics for 2019. https://www.domo.com/learn/data-never-sleeps-7. Last accessed 2020/07/04
9. Trabelsi, S., Plate, H., Abida, A., Aoun, M., Zouaoui, A., Missaoui, C., Ayari, A.: Mining social networks for software vulnerabilities monitoring. In: 2015 7th International Conference on New Technologies, Mobility and Security, pp. 1–7. IEEE (2015)
10. Imran, M., Castillo, C., Diaz, F., Vieweg, S.: Processing social media messages in mass emergency: a survey. ACM Comput. Surv. **47**(4), 1–38 (2015)
11. Kuzmina, N, Ridley, M.: Architecture of ontology construction and semantic search system for enhancing civil aviation processes. Sci. Bull. State Sci. Res. Inst. Civ. Aviation **28**, 103–113 (2019) (in Russian)
12. Kuzmina, N, Ridley, M.: Automatic domain ontology construction from text corpus for civil aviation information systems. Sci. Bull. State Sci. Res. Inst. Civ. Aviation **21**, 122–131 (2018) (in Russian)

Chapter 24
3D Object Classification, Visual Search from RGB-D Data

Vadim L. Kondarattsev, Alexander Yu. Kryuchkov, and Roman M. Chumak

Abstract In this chapter, we consider the problem of creating a system for processing 3D models obtained using RGB-D sensors for the purpose of semiautomatic selection and classification of objects and their auto-completion based on visual search. We have proposed several heuristic preprocessing algorithms for selecting an object of interest on a scan that contains noise and extraneous objects. To implement the visual search algorithm, we obtained a modification of the ray casting 3D-shape feature extraction algorithm. To solve the classification problem, the possibility of using deep learning architectures based on convolution mechanisms on graphs is investigated. The information about the object class obtained during the classification stage is used for faster and more accurate auto-completion. The resulting system has been tested on real data.

24.1 Introduction

In recent years, space scanning technologies using depth cameras and lidars have become widespread. This is due to the active development of such areas as autonomous vehicles, augmented and virtual reality, medical scanning, computer vision, and robotics. With an increasing number of 3D datasets and various tasks related to processing such data, creating systems for automatic detection and auto-completion of objects in 3D scenes becomes especially important.

V. L. Kondarattsev (✉)
Moscow Aviation Institute (National Research University), 4, Volokolamskoe shosse, Moscow 125993, Russian Federation
e-mail: vadim@phygitalism.com

V. L. Kondarattsev · A. Yu. Kryuchkov · R. M. Chumak
PHYGITALISM, 2/46 Bol'shaya Sadovaya ul., Moscow 123001, Russian Federation
e-mail: a.kryuchkov@phygitalism.com

R. M. Chumak
e-mail: p4@phygitalism.com

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4_24

In this chapter, we consider the problem of semiautomatic detection and classification of three-dimensional objects in three-dimensional scenes. Scene data is obtained by scanning the surrounding area using depth cameras. Detection (selection) of objects is performed using classical preprocessing methods: noise removal, deletion of supporting surfaces, and unnecessary objects. Deep learning methods are used to classify the object highlighted in the scene.

In Sect. 24.2, we present a comparative analysis of existing deep learning architectures designed for semantic segmentation of point clouds. Based on this analysis, we justify the choice of the LDGCNN architecture [1] for implementing the classification stage of the model. In Sect. 24.3, we consider the formal formulation of classification and search problems for 3D models (also named as meshes), and introduce all the necessary terms and mathematical constructions. In Sect. 24.4, we look at various aspects of data preprocessing. In particular, to highlight an object of interest on a 3D scene, we introduce heuristic algorithms for removing the floor and foreign objects. For the problem of classification using a neural network, we introduce an algorithm for preprocessing 3D models made by 3D artists, which allows us to diversify the input data and bring it closer to the real raw data obtained from scanners. Raw data, which is usually represented in the form of a point cloud, does not allow creating visualizations of desirable quality. Therefore, separately in Sect. 24.5, we considered the problem of the automatic completion of a 3D scene with objects from a database of polygonal 3D models. In order to find the closest-shaped model in the dataset, we will need to solve the problem of searching in the space of three-dimensional models. For this purpose, we have developed an algorithm based on the ray casting method to obtain the descriptive representation of 3D model [2]. In Sect. 24.6, we present conclusions from the work done and discuss further work on the application of the considered algorithms and their improvement.

24.2 Related Work

From the point of view of classical machine learning, the task of constructing such an algorithm is usually divided into two subtasks:

1. Selection of informative features from the entire description of the object.
2. Application of machine learning algorithm (classification, clustering, etc.) to the selected description.

The first subtask is traditionally based on a good understanding of the subject area and the specifics of data. Thus, for example, in [3], the authors obtained an effective algorithm for extracting informative features from a point cloud. Firstly, for each point from the point cloud, the optimal, in the sense of proximity by some metric, number of points is the nearest neighbors for the selected subset of points. Then the covariance matrix is calculated, and its own vectors are used to construct various information signs.

However, in recent years, with the rapid development of the field of deep learning, approaches based on the use of deep neural networks, which allow combining the stage of feature selection and classification into one algorithm, are becoming more popular. One of the ways to build taxonomy of deep architectures for processing a point cloud can be a method based on the division of architectures into classes depending on how the input data is processed. Thus, for example, in [4], authors divide all architectures into direct methods and indirect ones.

Indirect architectures do not process the initial point cloud, but some intermediate representation (it can be voxelized models or a set of point cloud images in RGB-D format). Most often, indirect methods are inferior in quality to direct ones, and besides, they are more expensive in terms of memory (you need memory to store intermediate forms of data) and of time (you need time to get an intermediate representation).

On other hand, the direct methods of deep learning for solving the problems of processing a point cloud, similar to the study [4], can be divided into different classes of methods, depending on the types of operators—hidden layers used in building of the architecture or depending on the modification of the basic architecture of deep learning, based on which specific methods have been developed.

In order to select a deep learning model for solving the point cloud classification problem, we performed a comparative analysis of quality metrics from various sources mentioned in [4]. The comparison results are shown in Table 24.1. The comparison was made for datasets [5–8]. The following metrics from [4] were considered for comparing models:

- Overall accuracy.
- Mean accuracy.
- Mean intersection over union.

As a result, the LDGCNN model [1] was chosen for the practical implementation in the automatic object classification system. This method is a direct method, and it is based on modification of the GCNN architecture [9].

24.3 Formal Statement of the Problem

Our global task is divided into three stages:

1. Preprocessing of the polygon models.
2. Classification of the selected polygon model.
3. Autocomplete based on a search among similar models of the same class that was defined at the previous stage.

Before proceeding to the description of the main stages, it is necessary to describe how this data is presented in a more formal form. The classification of polygonal models will be solved as the classification of the point cloud problem. At the moment, this solution is associated with the possibility of using a broader class of algorithms

Table 24.1 Comparison of metrics for deep learning models for semantic segmentation of point clouds

Architecture	Dataset	Overall accuracy	Mean accuracy	Mean intersection over union
SEGCloud	S3DIS	–	57.35	48.92
RSNet	S3DIS	–	59.42	51.93
RSNet	ScanNet	–	48.37	39.35
RSNet	ShapeNet-part	–	–	84.9
LDGCNN	ModelNet40	92.9	90.3	–
LDGCNN	ShapeNet-part	–	–	85.1
SpiderCNN	ModelNet40	92.4	–	–
SpiderCNN	ShapeNet-part	–	–	85.3
PointNet++	ModelNet40	90.7	–	–
PointNet++	ScanNet (with voxelization)	84.5	–	–
MVCNN	ModelNet40	90.1	–	–
VoxNet	ModelNet40	–	83	–
SO-Net	ShapeNet-part	–	–	84.6
SO-Net	ModelNet40	90.8	–	–
RGCNN	ShapeNet-part	–	–	84.3
RGCNN	ModelNet40	90.5	87.3	–
3DMAX-Net	S3DIS	79.5	–	47.5
PointSIFT	S3DIS	88.72	–	70.23
PointSIFT	ScanNet	86.2	–	41.5
PointGrid	ModelNet40	92.0	88.9	–
PointCNN	ModelNet40 (pre-aligned)	92.5	88.8	–
PointCNN	ModelNet40 (unaligned)	92.2	88.1	–
PointCNN	ScanNet	85.1	–	–
PointCNN	S3DIS	88.1	–	65.39
PointCNN	ShapeNet-part	–	–	84.6
GAPNet	ModelNet40	92.4	89.7	–
GAPNet	ShapeNet-part	92	84.7	–
A-CNN	ModelNet40	92.6	90.3	–
A-CNN	ScanNet	85.4	–	–
A-CNN	S3DIS	87.3	–	–
A-CNN	ShapeNet-part	86	–	–
3P-RNN	S3DIS	86.9	73.6	56.3

(continued)

Table 24.1 (continued)

Architecture	Dataset	Overall accuracy	Mean accuracy	Mean intersection over union
3P-RNN	ScanNet	76.5	–	–
DGCNN	ShapeNet-part	–	–	85.1
DGCNN	S3DIS	84.1	–	56.1

to solve the problem, although there are methods for processing polygon models directly. The classification of point clouds was chosen because actual scans from devices are often incomplete and subject to various distortions. As a result, the use of methods for classifying meshes becomes more difficult because the training models do not match the real examples.

Hereinafter, Sect. 24.3.1 discusses point cloud data and point clouds classification. Mesh data and meshes classification are described in Sect. 24.3.2. Searching among polygonal models is presented in Sect. 24.3.3.

24.3.1 Point Cloud Data and Point Clouds Classification

The original problem can be formulated as a classification problem for a set of points $P^{n,k} = \left\{x^i | x^i = \left(x_1^i, x_2^i \ldots, x_k^i\right)^T \in \mathbb{R}^k\right\}_{i=1}^{n}$, where n is the number of points and k is the dimension of the space. We need to find the function: $K = C\left(P^{n,k}\right)$, where $K = \{1, 2, \ldots, N_C\}$ and N_C is the number of classes. To solve the classification problem, we will use a neural network. As a result, the classification itself consists of several stages: $C\left(P^{n,k}\right) = (NC \circ N)\left(P^{n,k}\right)$, where $P'^{s,k} = N\left(P^{n,k}\right)$ is the preprocessing function and for $s \leq n$, $K = NC\left(P'^{s,k}\right)$ is the direct classification algorithm. Next, we will describe how each of these functions works.

24.3.2 Mesh Data and Meshes Classification

As a result of scanning, it is also possible to get a polygon model in the form $M_{n,m} = (V_n, F_m)$, where $F_m = \left\{f^l = \{i, j, k\} | x^i, x^j, x^k \in V_n, \left(x^i, x^j, x^k\right) - \text{form a polygon}\right\}$ is a set of faces, V_n is a set of vertices, m is the number of faces, $l \in \{1, 2, \ldots, m\}, i, j, k \in \{1, 2, \ldots, n\}$. We assume that all models are triangulated.

For a polygonal model, we introduce the concept of a surface in the same way as it was done in [2]. To do this, we first define the concept of the surface of an elementary triangular polygon as a set of points in the polygon plane bounded by the polygon faces:

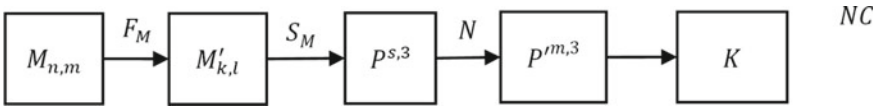


Fig. 24.1 Main preprocessing steps that must be performed to classify the object received from the scanner

$$T_l = \{v \in \mathbb{R}^3 | v = \alpha x^i + \beta x^j + (1 - \alpha - \beta)x^k, \alpha, \beta \geq 0, \alpha + \beta \leq 1\},$$

where $f^l = \{i, j, k\} \in F_m, x^i, x^j, x^k \in V_n$. Then, we will define the surface I of the polygon model $M_{n,m} = (V_n, F_m)$ as the union of polygon surfaces that this model consists of: $I(M_{n,m}) = \bigcup_{i=1}^{m} T_i$.

We shall say that axis-aligned bounding box (AABB) of the polygon model is a set of eight points in space:

$$B(M_{n,m}) = \{x^1_{\min}, x^1_{\max}\} \times \{x^2_{\min}, x^2_{\max}\} \times \{x^3_{\min}, x^3_{\max}\},$$
$$x^i_{\min} = \min_{x \in V_n} x_i, x^i_{\max} = \max_{x \in V_n} x_i.$$

Volume of the bounding box we denote as:

$$\mathrm{Vol}_{bb}(M_{n,m}) = \prod_{i=1}^{3} (x^i_{\max} - x^i_{\min}).$$

Geometric center c_g of the bounding box is a point in space of the following type:

$$c_g(M_{n,m}) = \frac{1}{2}(x^1_{\min}, x^2_{\min}, x^3_{\min})^T + \frac{1}{2}(x^1_{\max}, x^2_{\max}, x^3_{\max})^T.$$

The $C_M(M_{n,m})$ classification problem is reduced to the $C(P^{s,3})$ classification problem, where $P^{s,3} = (S_M \circ F_M)(M_{n,m})$, $M'_{k,l} = F_M(M_{n,m})$, $M'_{k,l} = (V'_k, F'_l)$ is a polygon model after filtering (with removed isolated polygons, removed individual vertices, etc.), and $P^{s,3} = S_M(M'_{k,l})$ is a function for converting mesh to point cloud.

All stages can be represented as a diagram (see Fig. 24.1).

24.3.3 Searching Among Polygonal Models

Let $O = \{M_{k_i,l_i}\}_{i=1}^{n}$ be a set of polygonal models, where n is a number of models. In our case O is a kind of database with polygonal models. We assumed that a measure

of difference is defined for any two models:

$$d = \rho\left(M_{p,s}, M_{k,l}\right) \in [0; 1].$$

The difference measure is based only on the model geometry. The more dissimilar the models are in shape, the greater the d value. Our goal is to find the set of models $R_r\left(M_{p,s}, O\right) \subset O, r = 1, 2, \ldots, n$, where elements of the set are the most similar models for the source model $M_{p,s}$. We will construct this set in the following way:

$$R_r\left(M_{p,s}, O\right) = \bigcup_{i=1}^{k} R^i,$$

$$R^i = \arg \min_{M \in O / \bigcup_{l=1}^{i-1} R^l} \rho\left(M_{p,s}, M\right),$$

$$\bigcup_{i=1}^{0} R^i = \emptyset,$$

$$k = \min_{k \in \{1,2,\ldots,n\}} \left\{ k : \left| \bigcup_{i=1}^{k} R^i \right| \geq r \right\}.$$

If the number of models in union $\bigcup_{i=1}^{k} R^i$ is less than r, then we will randomly choose $r - \left|\bigcup_{i=1}^{k} R^i\right|$ models from the set R^k.

24.4 Data Preprocessing

In this section, preprocessing of polygonal models is considered in Sect. 24.4.1, while point cloud classification is described in Sect. 24.4.2.

24.4.1 Preprocessing of Polygonal Models

For more clarity and tests on real data, we scanned a small area of the room using a depth camera. The result of scanning in the form of a polygon model is shown in Fig. 24.2.

We assumed that the object of interest (table) is located near of bounding boxes geometric center. Here, under the bounding box, we understand AABB of the entire scan.

If we look at the source data, we can notice several things that prevent us from selecting the desired object:

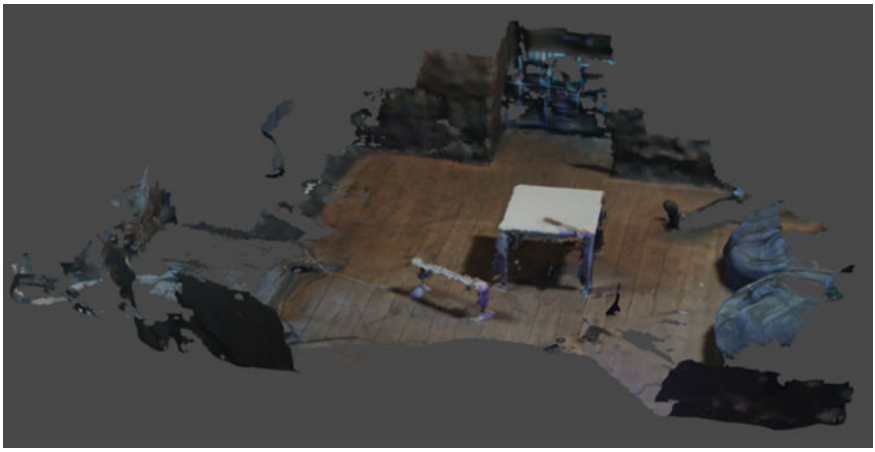


Fig. 24.2 Input scan as polygonal model

- Noise. All sensors have an error that leads to the appearance of various artifacts. For example, isolated polygons and vertexes that hangs in the air.
- Floor and walls. The object we are interested in is standing on some kind of support surface. It is usually not needed for further analysis.
- Presence of foreign objects. For example, on the scan above (see Fig. 24.2), we can see scanned foreign objects (a heater, parts of bags, etc.) that are not required for further analysis of the object of interest.

We need to complete several stages of raw data preprocessing before we can select the object we are interested in.

The removal of the support surface. To remove the reference surface, we proposed a heuristic algorithm and implemented it using the Blender Python API. The main idea that the algorithm is based on is that polygons belonging to the floor are located at a low height, and their normals are not strongly deviated from the horizontal direction vector. The input to the algorithm is a polygon model $M_{n,m} = (V_n, F_m)$ and parameters α, β. The algorithm consists of the following steps:

1. Let $D = \emptyset$ be the set of polygons that belong to the floor (the reference surface).
2. Let us set the parameters $\alpha, \beta \in (0; 1)$, where α determines the fraction of the height of AABB, at which the reference surface can be located, and $\beta = 1 - \cos\gamma$, where γ is a maximum angle of deviation of the polygon normal from the vector $up = (0, 0, 1)^T$, which shows the "up" direction.
3. Compute AABB: $B\left(M_{n,m}\right)$ of polygonal model $M_{n,m} = (V_n, F_m)$.
4. Compute the height of AABB: $h = x^3_{\max} - x^3_{\min}$.
5. Compute $h_{\max} = \alpha \times h$.
6. For each polygon $p^g = \{r^g, t^g, s^g\} \in F_m$, $g = 1, 2, \ldots, m$:
   a. Compute point $\overline{x} = \frac{\widehat{x^{r^g}} + \widehat{x^{t^g}} + \widehat{x^{s^g}}}{3}$, where

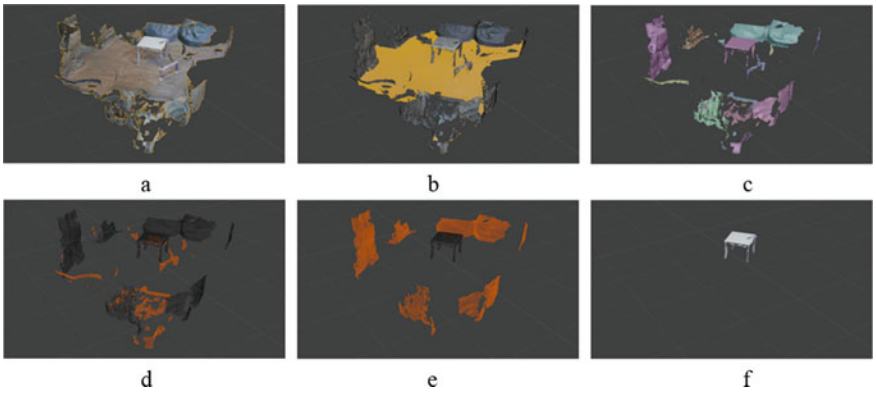


Fig. 24.3 3D scene on different stages of preprocessing: **a** original scan, **b** floor detection, $\alpha = 0.3$, $\beta = 0.2$, **c** connected component, **d** objects with too small number of vertices, $\alpha = 0.05$, **e** objects that are too far from center of bounding box, $\beta = 0.2$, and **f** object of interest

b. Compute polygons normal $n^g = \left[\widehat{x^{t^g}} - \widehat{x^{r^g}}, \widehat{x^{s^g}} - \widehat{x^{t^g}}\right]$, where $[\cdot, \cdot]$ denotes a vector product.
c. Let 'us make this normal of unit length: $\hat{n}^g = n^g / \|n^g\|$, where $\|\cdot\|$ denotes Euclidian norm.
d. If $\overline{x}_3 - x^3_{\min} \leq h_{\max}$ and $\left|n^g_3\right| \geq 1 - \beta$, then we add p^g to the set D: $D = D \cup \{p^g\}$.

7. Remove vertices and polygons of D set out of initial model $M_{n,m} = (V_n, F_m)$. We will get a new model $M'_{e,r} = \left(V'_e, F'_r\right)$, where $F'_r = F_m / D$, $r = m - |D|$, $V'_e = \bigcup\limits_{f \in F'_r} \left\{x^l | x^l \in V'_n\right\}_{l \in f}$ and e is a number of vertices in model after removing floor polygons.

The result of applying the algorithm described above on a real scan is shown in Fig. 24.3.

Removing noise and foreign objects. To remove noise polygons and foreign objects, we proposed a heuristic algorithm and implemented it using the Blender Python API. There are several basic ideas that this algorithm is based on: first, the object of interest is most often detailed and contains the largest number of polygons on the scan, and, second, under the assumption that the object of interest is close to the geometric center of the scan (in the sense of a bounding box), we can delete other objects that are not related to the object of interest and are located further from the geometric center. The input to the algorithm is a polygon model $M_{n,m} = (V_n, F_m)$ and parameters α, β. The algorithm consists of the following steps:

1. Compute AABB—$B\left(M_{n,m}\right)$ of polygonal model $M_{n,m} = (V_n, F_m)$.
2. The model is divided into connected components. Each component is a set of connected polygons. We can consider the polygon model $M_{n,m} = (V_n, F_m)$ as a spatial undirected graph G with a set of vertices coinciding with V_n, and the set of

edges is defined by the vertices belonging to polygons, i.e., if two vertices belong to the same polygon, then there is an edge in the graph between them. Then, the set of connected polygons is formed as the set of connectivity components of an undirected graph G [10]:

$$M_{n,m} = (V_n, F_m) = \bigcup_{i=1}^{k} M_{n_i,m_i} = \bigcup_{i=1}^{k} \left(V_{n_i}, F_{m_i}\right),$$
$$F_{m_i} \cap F_{m_j} = \emptyset, i, j = 1, \ldots, k, i \neq j,$$

where k is the number of connectivity components of graph G, V_{n_i} are the vertices that belong to the connectivity component with number i, F_{m_i} are the polygons that are formed by vertices from V_{n_i}.

3. For each set of vertices V_{n_i} of model $M_{n_i,m_i} = \left(V_{n_i}, M_{m_i}\right)$, we compute its geometrical median m_i [11]:

$$m_i = \arg \min_{y \in \mathbb{R}^3} \sum_{x \in V_n} \|y - x\|, i = 1, \ldots, k.$$

4. Let us set the parameter $\alpha \in [0; 1]$ is the ratio of the maximum assumed number of vertices in noise objects to the number of scan vertices, $\beta \in [0; 1]$ is the ratio of the minimum distance of the geometric median from the center of the bounding box $B\left(M_{n,m}\right)$.
5. Compute $d^{\max} = \left(\frac{x^1_{\max} - x^1_{\min}}{2}\right)^2 + \left(\frac{x^2_{\max} - x^2_{\min}}{2}\right)^2$.
6. Let 'us create a set of objects of interest $S = \emptyset$.
7. For each model $M_{n_i,m_i} = \left(V_{n_i}, F_{m_i}\right)$ with $i = 1, \ldots, k$:

   a. If $n_i > \alpha \cdot n$ and $\frac{\left(m_1 - d_1^{\max}\right)^2 + \left(m_2 - d_2^{\max}\right)^2}{d^{\max}} \leq \beta$, then we add M_{n_i,m_i} to the set S: $S = S \cup \left\{M_{n_i,m_i}\right\}$.

8. We will get a new model $M'_{k,l} = \left(V'_k, F'_l\right)$, where $V'_k = \bigcup_{(V,F) \in S} V$, $F'_l = \bigcup_{(V,F) \in S} F$.

The result of applying this algorithm to the scan with the removed floor obtained at the previous stage is shown in Fig. 24.3.

As a result, the F_m function consists of algorithms: removing the reference surface and removing noise along with extraneous objects. The implementation of these two algorithms is available on our GitHub page [12].

24.4.2 Point Cloud Classification

As mentioned above, we chose the implementation of the LDGCNN architecture to classify the selected model. The model was trained from scratch based on the Model Net 40 dataset, which, like most other 3D datasets, consists of models drawn by 3D

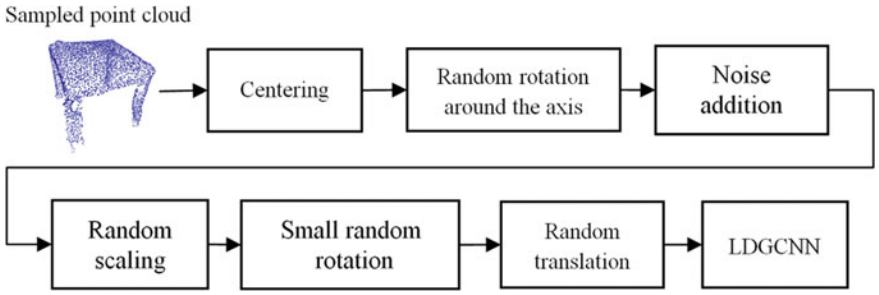


Fig. 24.4 Main steps scheme of preprocessing for neural network learning process

artists rather than scanned. In order for the trained model to better cope with the classification of real objects, the data on which it is trained must be modified.

Data preprocessing for the neural network learning process consisted of the following steps:

1. Converting a polygon model to a point cloud with a fixed number of points.
2. Initial scaling of the point cloud.
3. Random rotation around the axis that sets the direction "up".
4. The addition of noise.
5. Random scaling.
6. Small random rotation.
7. Random translation.

All these steps define N function and schematically represented in Fig. 24.4. The NC function is reduced to using the LDGCNN architecture to define the object class.

Converting polygonal models to point clouds. At this stage, we need to define the function S_m, which converts the model $M_{n,m}$ to a point cloud $P^{k,3}$. During the model training process, we fixed the number of points in the point clouds with value $k = 1024$. Source data with point clouds for training was stored in HDF5 format. From the scan, the results were converted using Open3D for Python [13]. Example of point cloud sampling with $k = 1500$ is shown on Fig. 24.5.

Initial scaling. This transformation of the point cloud is arranged so that each point is located after the transformation in a unit sphere. To do this, we first need to calculate AABB $-B\left(P^{k,3}\right)$. The scaling factor s is calculated as follows:

$$s = 2 \cdot \left\| \left(x^1_{\max}, x^2_{\max}, x^3_{\max}\right)^T - \left(x^1_{\min}, x^2_{\min}, x^3_{\min}\right)^T \right\|^{-1}.$$

As a result, we get a point cloud of the form:

$$P^{k,3}_c = \left\{x' | x' = s \times x, x \in P^{k,3}\right\}.$$

Random rotation around "up" axis. This operation is necessary to get a new point cloud rotated around the y axis. We will define this direction as "up" in our

Fig. 24.5 Conversion mesh to point cloud with $k = 1500$

work. As a result, we get a point cloud of the form:

$$P_r^{k.3} = \left\{ x' | x' = \begin{pmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{pmatrix} x, x \in P_c^{k,3} \right\},$$

where $\phi \sim U(0, 2\pi)$ is a random variable with a uniform distribution on $[0, 2\pi]$.

Noise addition. This step is necessary because in practice the data from the sensors are subject to various distortions. Because of this, the geometry of objects is not so "perfect" compared to the examples in the training dataset. For the stability of the classification, a small noise is artificially added to the training examples. As a result, we get a point cloud of the following type:

$$P_j^{k.3} = \left\{ x' | x' = x + (\varepsilon_1, \varepsilon_2, \varepsilon_3)^T, x \in P_r^{k,3} \right\},$$

where $\varepsilon_i \sim N\left(0, \sigma^2\right)$ is the independent random variables with normal distribution, mathematical expectation equal to zero and variance equal to σ^2. During network training, the following distribution parameters were set: $\sigma = 0.01$, If $\varepsilon_i > 0.05$, then it was fixed equal $\varepsilon_i = 0.05$, if $\varepsilon_i < -0.05$, then it was fixed equal to $\varepsilon_i = -0.05, i = 1, 2, 3$.

Random scaling. This operation is necessary to get a new point cloud of the form:

$$P_s^{k.3} = \left\{ x' | x' = s \times x, x \in P_j^{k,3} \right\},$$

where $s \sim U(a, b)$. During network training, the following parameters were set: $a = 0.8, b = 1.25$.

Small random rotation. This operation is necessary to get a new point cloud that is rotated at small angles relative to different axes:

$$P_{rr}^{k.3} = \left\{x' | x' = R_z \times R_y \times R_x \times x, x \in P_s^{k,3}\right\},$$

$$R_x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \sin\phi_1 & -\sin\phi_1 \\ 0 & \sin\phi_1 & \cos\phi_1 \end{pmatrix}, R_y = \begin{pmatrix} \cos\phi_2 & 0 & \sin\phi_2 \\ 0 & 1 & 0 \\ -\sin\phi_2 & 0 & \cos\phi_2 \end{pmatrix},$$

$$R_z = \begin{pmatrix} \cos\phi_3 & -\sin\phi_3 & 0 \\ \sin\phi_3 & \cos\phi_3 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

where R_x, R_y, R_z are the rotation matrices around the main axes, ϕ_i is the independent random variables with normal distribution $\phi_i \sim N\left(0, \sigma^2\right)$. During network training, the following parameters were set: $\sigma = 0.06$, if $\phi_i > 0.18$, then it fixed equal to $\phi_i = 0.18$, if $\phi_i < -0.18$, then it fixed equal to $\phi_i = -0.18$, $i = 1, 2, 3$.

Random translation. This operation consists of randomly translating each polygonal model in the training dataset in parallel:

$$P_j^{k.3} = \left\{x' | x' = x + c, x \in P_{rr}^{k,3}\right\},$$

where $c = (\varepsilon_1, \varepsilon_2, \varepsilon_3)^T$, $\varepsilon_i \sim U(a, b)$, $i = 1, 2, 3$ are the independent random variables with a uniform distribution on $[a, b]$. During network training, the following parameters were set: $a = -0.1$, $b = 0.1$. A fixed vector was generated for each example during the training process.

Model fitting. Training the model on the NVIDIA RTX 2080Ti GPU took approximately 7 h. 250 epochs were completed for training the main model and 100 epochs for training the classifier. LDGCNN is a model that needs to be trained in two stages: training the main part to extract features and training the classifier with a fixed model for features. The implementation of LDGCNN, which is available on GitHub [14], was used as a basis. The average class accuracy [15] value on the test set is equal to 0.900255. An accuracy evaluating graph during training of the feature extraction part is shown in Fig. 24.6.

24.5 Ray Casting Descriptive Representation of 3D Models

To make it easier to work with 3D model after classification, the actual scanned object can be replaced with 3D model from the database of ready-made models (in this sense, we will assume the procedure of 3D scene auto-completion). Of course, there may not be an exact copy of the scanned model, but if the model from the database is close enough in terms of surface shape and class, then working with such a model can be much more convenient.

There are different ways to organize the search procedure for different types of data. One of such approaches is the search based on a descriptive representation of

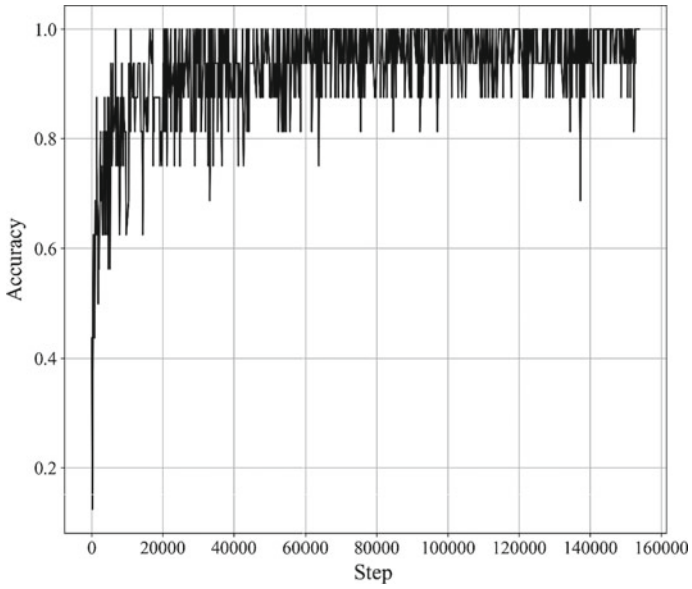


Fig. 24.6 Train set accuracy evaluation

3D models. In this chapter, we propose our own heuristic algorithm for the descriptive representation of 3D models based on the ray casting method and a method for comparing such descriptive representations. There are also many other ways to construct descriptors for 3D models. For example, a whole group of methods is based on calculating the moment characteristics of a polygon mesh. The most relevant work devoted to the construction of moment descriptors is [16], in which authors used three-dimensional Zernike moments. Comprehensive information about methods of constructing descriptive representations (descriptors) for 3D models can be found in the doctoral dissertation [2].

In order to describe the descriptors constructing algorithm, let 'us first introduce all the necessary operators and concepts in Sect. 24.5.1. Description and search are discussed in Sect. 24.5.2. Section 24.5.3 provides a description of experiments.

24.5.1 Constructions and Operator

Rays whose lengths will be used to construct a descriptive representation of the mesh will be emitted from points of the unit icosphere $\text{Ico}_{n,m} = \left\{V_n^{\text{ico}}, P_m^{\text{ico}}\right\}$, where V_n^{ico} are the icosphere mesh vertices and P_m^{ico} are the icosphere mesh polygons [17]. Icosphere's main feature, which we will use, is that all its vertices belong to the surface of a two-dimensional sphere $x^i \in S_2$ (S_2 is the surface of a two-dimensional sphere).

To ensure that the model processing does not depend on the method of triangulation or orientation of the model, the choice of the model center that will be combined with the center of the icosphere also should not depend on the method of triangulation and orientation of the model. As the center of the model, we will use the center of the minimum volume bounding box [18], which we denote as $c_{\text{vol}}\left(M_{n,m}\right)$.

Polygonal model $M_{n,m}$ need to be scaled to fit into a unit icosphere. To do this, let 'us determine the scaling factor:

$$l_s\left(M_{n,m}\right) = \frac{1}{\max\limits_{x \in V_n} \|x\|}.$$

On the surface of the icosphere, we define a finite number of unit vectors oriented to the center:

$$u_i = \frac{-x^i}{\left\|x^i\right\|}, x^i \in I\left(\text{Ico}_{n,m}\right).$$

In case when we have a polygon model surface $I\left(M_{n,m}\right)$ placed inside a unit icosphere by scaling and translation, it is possible for each vector u and space line, that this guide vector sets, define a vector R consisting of real numbers obtained from the distances between origin of the u and points of intersection that line with the surface of polygonal model. More formally, this vector can be defined:

$$\begin{aligned} R(u, I) &= \left(r^1, r^2, \ldots, r^M\right), \\ r^i &\in [0, 2], r^i \cdot u \in I \cup v^I, \\ i &\in \{1, 2, \ldots, M\},\ 0 \le r^i \le r^{i+1} \le 2\}, \end{aligned}$$

where v^I is the vertex of the icosphere opposite the origin of vector u, and M is the number of points where the ray intersects with the model surface. Note that this vector always exists and consists of at least one element—the distance to the opposite vertex of the icosphere, i.e., $r^M = 2$. In the algorithm described below, we will construct this set for all selected vertices of the icosphere $v_i \in \text{Ico}_{n,m}$ and for the opposite vertices $v_i' \in \text{Ico}_{n,m}$ (see Fig. 24.7).

24.5.2 Descriptization and Search

Usually, a descriptive representation is a vector [2], but in our case it will be a matrix $D_{4\times N} \in R^{4\times N}$, which we will construct from N four-dimensional vectors:

$$D = (d_1|d_2|\ldots|d_N), \quad d_i = \left(d_i^1, d_i^2, d_i^3, d_i^4\right)^T \in \mathbb{R}^4. \tag{24.1}$$

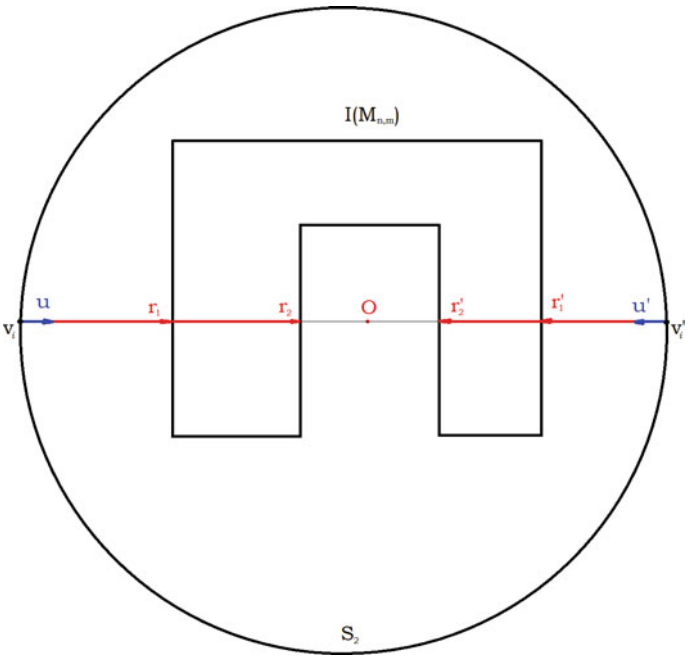


Fig. 24.7 Constructing R set for opposite icosphere vertices v_i and v_i'

Let us describe the algorithm for converting a polygonal model $M_{n,m}$ in its descriptive representation $D(M_{n,m})$.

Constructing the descriptors. The input of the algorithm is a polygonal model $M_{n,m}$ with its surface $I(M_{n,m})$, number of vertices of icosphere is equal to $2N$. As a result, the algorithm returns descriptive representation of the model as a matrix $D_{4\times N}$.

Steps of the algorithm are the following:

1. The model is placed inside a unit icosphere by scaling and sequentially combining the origin of coordinates with the geometric center of the model's minimum volume bounding box:

$$M'_{n,m} = \{V'_n, F_m\}, \text{где } V'_n = \{x' : x' = x - c_{vol}(M_{n,m}), x \in V_n\};$$

$$M''_{n,m} = \{V''_n, F_m\}, \text{где } V''_n = \{x'' : x'' = l_s(M'_{n,m}) \cdot x', x' \in V'_n\}.$$

2. All vertices of the icosphere form N pairs of opposite vertices $v_i, v'_i, i = 1, \ldots, N$, which corresponds to the guide vectors u_i, u'_i. Sets of ray lengths are calculated for each such pair:

$$R(u_i, I) = (r_i^1, r_i^2, \ldots),$$

$$R\left(u_i', I\right) = \left(r_i'^1, r_i'^2, \ldots\right).$$

Vector d_i is constructed in the form:

$$d_i = \left(r_i^1, r_i^2, r_i'^1, r_i'^2\right).$$

3. Descriptor matrix D formed from vectors d_i:

$$D = \left(d_{i_1} \middle| d_{i_2} \middle| \ldots \middle| d_{i_N}\right), \quad \text{where} \forall i_j, i_{j+1}, d_{i_j}^1 \leq d_{i_{j+1}}^1.$$

Let us make some comments about the algorithm described above:

- Because of the sorting procedure in step 3 of Algorithm 1, it becomes impossible to reconstruct the model from its descriptive representation, because there are exactly $N!$ ways to order the element set, which means that one descriptive representation of the form of Eq. 24.1 corresponds to several possible polygonal models.
- By entering into consideration, the distance not only to the nearest intersection point with the model, as is done in [19], but also to the next intersection points with the models surface, it is possible to achieve that polygonal models that have the same external surface shape, but different internal structure will have different descriptors.
- To find out the lengths of vectors r^i, it is possible to use a computationally efficient ray procedure with a plane from the work [20].

Next, we will construct an algorithm that allows us to calculate the similarity measure between two descriptive representations described above. This measure similarity is a function of the following type:

$$\mu(D_1, D_2): \mathbb{R}^{4\times N} \times \mathbb{R}^{4\times N} \to [0, 1].$$

In order to design a search algorithm based on the similarity of descriptive representations in the future, it will be sufficient to use the metric ρ in the descriptor space defined earlier in Sect. 24.3.3, let us define the distance function of the following type:

$$\rho\left(M_{p,s}, M_{k,l}\right) = 1 - \mu\left(D_N\left(M_{p,s}\right), D_N\left(M_{k,l}\right)\right),$$

where $M_{p,s}, M_{k,l}$ are two polygonal models and $D_N\left(M_{p,s}\right), D_N\left(M_{k,l}\right)$ are their descriptive representations in the form of Eq. 24.1 with dimensions shape $4 \times N$. For convenience of notation, two different descriptive representations will be written as:

$$D_1 = \left(d^{11} \middle| d^{12} \middle| \ldots \middle| d^{1N}\right);$$
$$D_2 = \left(d^{21} \middle| d^{22} \middle| \ldots \middle| d^{2N}\right).$$

Similarity measure of descriptors. The input of the algorithm is a descriptive representations of two models D_1, D_2, proximity threshold value $\varepsilon \geq 0$. As a result, the algorithm returns similarity measure $\mu(D_1, D_2)$ between two descriptors.

Steps of the algorithm:

1. Let Ω be a set of vectors from the second descriptive representation D_2, that have not yet participated in the comparison process. At the beginning, this set coincides with the set of vectors that the second descriptor consists of $\Omega = \{d^{21}, d^{22}, \ldots, d^{2N}\}$.
2. Also enter into consideration n that is the number of vectors for two descriptors that are close to each other in the sense of the metric:

$$\rho'(x^1, x^2) = \sum_{i=1}^{4} |x_i^1 - x_i^2|, x^1, x^2 \in R^4.$$

   At the beginning of the algorithm $n = 0$.
3. Let 'us find the number of vectors pairs from two descriptors that lie no further apart than the threshold distance. To do this, we will iterate through all the vectors d^{1i} in the first descriptor D_1.
   - 3.1 For each d^{1i} we will iterate over vectors from the set $d^{2j} \in \Omega$.
   - 3.2 Calculate the distance $\rho'(d^{1i}, d^{2j})$.
   - 3.3 If this distance does not exceed the threshold value: $\rho' < \varepsilon$, then the vector $d^{2j} : \Omega = \Omega \backslash \{d^{2j}\}$ is excluded from further comparison, then we increase the counter $n = n + 1$, otherwise, the set Ω traversal continues.
4. Iterating over all vectors from D_1, we can calculate the similarity measure as a ratio:

$$\mu = \frac{n}{N}.$$

Despite the fact that the algorithm was obtained empirically, in practice it shows an acceptable quality of work.

24.5.3 Experiments

In our tests, we fix the number of vertex pairs on the icosphere equal to $N = 640$. This number of points for emitting rays from the icosphere surface allows to achieve an acceptable execution speed of the algorithm with the search quality already stabilized. When conducting tests, it was experimentally checked that the ranked output of the search stops changing with increasing of N. Examples of such tests for various N values are shown in Fig. 24.8.

The implementation of the algorithm is written in Python 3 and Rust programming languages and is released as an add-on for Blender 3D software.

Fig. 24.8 For a subset of models (1000 models per category for 4 categories), we compared the ranked results for different values of $2 \cdot N$ (left column)

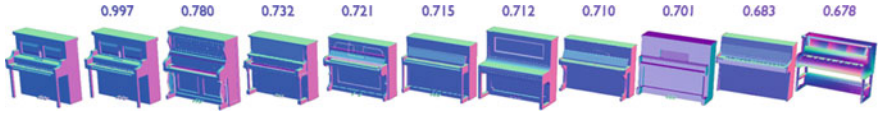


Fig. 24.9 Example of ranked output in the task of searching for the most similar shaped objects from the ShapeNet dataset: the leftmost object is a search sample taken from the dataset itself, and the other objects are ordered from left to right as the similarity index μ decreases (the value signed above the objects)

Besides descriptive representations, there are other ways to compare 3D models. For example, the distance function can be defined directly between two polygon models. Similar to how it is done in [21], we will consider the chamfer distance and the normal distance functions. For each ranked output of the search algorithm, we will additionally calculate the values of these two distances between the object being searched for and all other objects in the ranked output.

The results of the search algorithm are shown below. Figure 24.9 shows a ranked search output for the closest-shaped objects from the ShapeNet dataset. The sample object is a model from the dataset itself. It is notable that the closest model in terms of similarity measures in the ranked category coincides with the model sample.

Figure 24.10 shows the values of chamfer loss and normal loss for ranked search results from Fig. 24.9. It can be noted that despite the fact that all objects in the search results are oriented in the same way and have the same scale, the indicators of the loss functions do not allow to draw any qualitative conclusions about the search results, which shows the need for the descriptive approach usage.

Figure 24.11 shows a ranked search output for the closest-shaped objects from the ShapeNet dataset, where the sample object is a scanned object after preprocessing stages. Figure 24.12 shows the values of chamfer loss and normal loss for ranked output from Fig. 24.11.

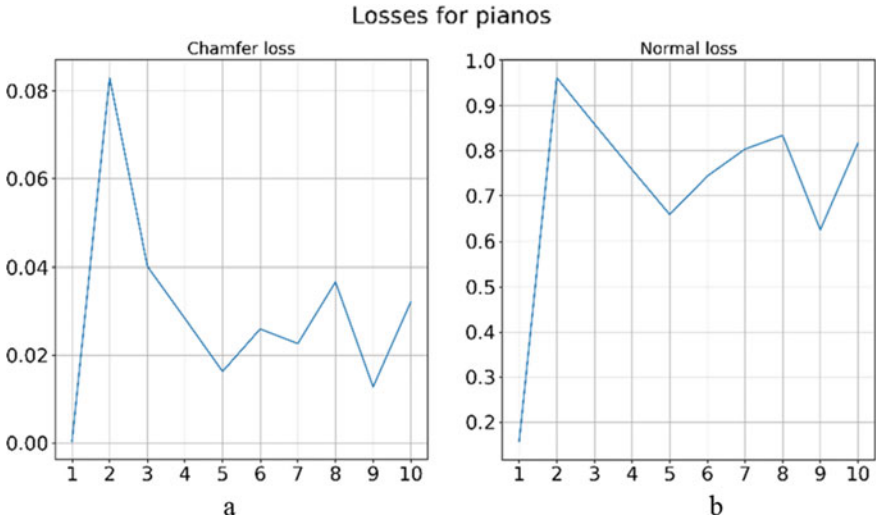


Fig. 24.10 Losses of pianos: **a** values of chamfer loss, **b** values of normal loss for objects from the previous image (see Fig. 24.9). On the x-axis there are the objects indexes in the ranked output

Fig. 24.11 Example of ranked output in the task of searching for the most similar-shaped objects from the ShapeNet dataset: the leftmost object is a sample for searching, obtained as a result of 3D scanning and selected as an object of interest, the remaining objects are ordered from left to right as the similarity index μ decreases (the value signed above the objects)

Figure 24.13 shows a situation where knowing the object class to search for similar objects allows to achieve more correct results: when auto-completing a model from the database, it will be replaced with the model of the right class.

24.6 Conclusions

This chapter demonstrates the possibility of using deep learning methods to create a system for automatic 3D scanned objects classification. The choice of the appropriate deep architecture is based on a comparative analysis of existing SOTA models executed on different datasets.

To select an object of interest from a large-scale space scan, we considered preprocessing methods: noise data filtering, reference planes deletion, and removing extraneous objects. An algorithm for descriptive representation of three-dimensional

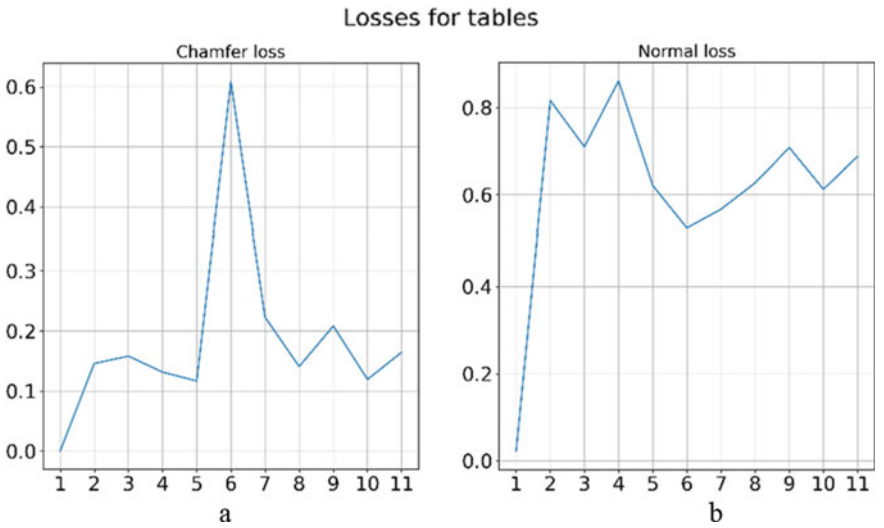


Fig. 24.12 Losses for tables: **a** values of chamfer loss, **b** values of normal loss for objects from the previous image (see Fig. 24.11). On the x-axis there are the objects indexes in the ranked output

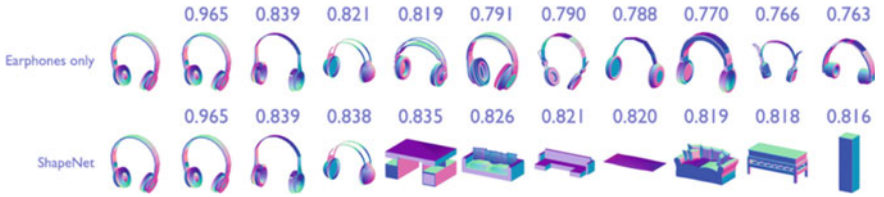


Fig. 24.13 Search algorithm result for the headphone model: among the "headphones" class of models (top row) and among all models from ShapeNet dataset (bottom row). The first model in both rows is the sample one. Above the models there are values of the similarity measure μ

models based on modification of existing methods of ray casting is obtained. The possibility of using this descriptive representation to solve the problem of searching among 3D models and using search results for 3D scene auto-completion is demonstrated.

The system constructed from all the parts considered allows to automatically classify data obtained from various scanners, search for duplicates within 3D datasets and to design beautiful scenes by replacing objects with the similar but more qualitative ones from 3D models collections as well.

In the future, we plan to improve the developed algorithms for description and 3D models search and try to use the resulting system for construction and analysis of the 3D scenes semantic graphs [22]. Representation of spatial scenes by their semantic graphs is a new actively developing area of 3DML that will allow us to better

understand the geometric structure of the surrounding space and its relationships to semantic information about objects.

References

1. Zhang, K., Hao, M., Wang, J., de Silva, C.W., Fu, C.: Linked dynamic graph CNN: learning on point cloud via linking hierarchical features. CoRR ArXiv Preprint, arXiv:1904.10014 [cs.CV] (2019)
2. Vranic, D.V.: 3D model retrieval. Ph. D. Dissertation, University of Leipzig (2004)
3. Weinmann, M., Jutzi, B., Hinz, S., Mallet, C.: Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. ISPRS J. Photogramm. Remote Sens. **105**, 286–304 (2015)
4. Zhang, J., Zhao, X., Chen, Z., Lu, Z.: A review of deep learning-based semantic segmentation for point cloud. IEEE Access **7**, 179118–179133 (2019)
5. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3D ShapeNets: a deep representation for volumetric shapes. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1912–1920. IEEE Computer Society Press, Boston, MA, USA (2015)
6. Armeni. I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S.: 3D semantic parsing of large-scale indoor spaces. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognit., pp. 1534–1543. IEEE Computer Society Press, Las Vegas, NV, USA (2016)
7. Yi, L., Shao, L., Savva, M., Huang, H., Zhou, Y., Wang, Q., Graham, B., Engelcke, M., Klokov, R., Lempitsky, V., Gan, Y., Wang, P., Liu, K., Yu, F., Shui, P., Hu, B., Zhang, Y., Li, Y., Bu, R., Sun, M., Wu, W., Jeong, M., Choi, J., Kim, C., Geetchandra, A., Murthy, N., Ramu, B., Manda, B., Ramanathan, M., Kumar, G., Preetham, P., Srivastava, S., Bhugra, S., Lall, B., Haene, C., Tulsiani, S., Malik, J., Lafer, J., Jones, R., Li, S., Lu, J., Jin, S., Yu, J., Huang, Q., Kalogerakis, E., Savarese, S., Hanrahan, P., Funkhouser, T., Su, H., Guibas, L.: Large-scale 3D shape reconstruction and segmentation from ShapeNet Core55. CoRR ArXiv Preprint, arXiv:1710.06104 [cs.CV] (2017)
8. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2432–2443. IEEE Computer Society Press, Honolulu, HI, USA (2017)
9. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph Cnn for learning on point clouds. ACM Trans. Graph. **38**(5), 146.1–146.13 (2019)
10. Reingold, O.: Undirected connectivity in log-space. J. ACM. **55**, 1–24 (2008)
11. Cohen, M.B., Lee, Y.T., Miller, G., Pachocki, J., Sidford, A.: Geometric median in nearly linear time. In: The 48th Annual ACM SIGACT Symposium on Theory of Computing—STOC 2016, pp. 9–21. ACM Press, New York, New York, USA (2016)
12. GitHub page of our project. https://github.com/phygitalism/blender_scan_filter. Last accessed 2020/06/26
13. Zhou, Q.-Y., Park, J., Koltun, V.: Open3D: A modern library for 3D data processing. CoRR ArXiv Preprint, arXiv:1801.09847 [cs.CV] (2018)
14. GitHub page of LDGCNN. https://github.com/KuangenZhang/ldgcnn. Last accessed 2020/06/26
15. Kelleher, J.D., Namee, B.M., D'Arcy, A.: Fundamentals of Machine Learning for Predictive Data Analytics Algorithms: Worked examples, and Case Studies. The MIT Press, London (2015)
16. Novotni, M., Klein, R.: 3D Zernike descriptors for content based shape retrieval. In: Symposium on Solid Modeling and Applications, pp. 216–225. Washington, USA (2003)
17. Ede, J.D., Wenninger, M.J.: Spherical models. Math. Gaz. **65**(431), 65 (1981)

18. Toussaint, G.T.: Solving geometric problems with the rotating calipers. In: IEEE MELECON'83, p. A10. Athens, Greece (1983)
19. Vranic, D.V.: An improvement of ray-based shape descriptor. In: 8th Leipziger Informatics, pp. 55–58. Leipzig, Germany (2000)
20. Moller, T., Trumbore, B.: Fast, minimum storage ray/triangle intersection. ACM J. Graph. Tools. **2**(1), 21–28 (1997)
21. Gkioxari, G., Malik, J., Johnson, J.: Mesh R-CNN. In: IEEE International Conference on Computer Vision, pp. 9785–9795. Seoul, Korea (2019)
22. Fisher, M., Savva, M., Hanrahan, P.: Characterizing structural relationships in scenes using graph kernels. ACM Trans. Graph. **30**(4), 34.1–34.11 (2011)

Author Index

L. C. Jain et al. (eds.), *Applied Mathematics and Computational Mechanics for Smart Applications*, Smart Innovation, Systems and Technologies 217,
https://doi.org/10.1007/978-981-33-4826-4