

3

МОДЕЛИРОВАНИЕ И АНАЛИЗ ДАННЫХ

НАУЧНЫЙ ЖУРНАЛ

MODELLING AND DATA ANALYSIS

SCIENTIFIC JOURNAL

2020

ISSN: 2219-3758
ISSN: 2311-9454 (ONLINE)

МОДЕЛИРОВАНИЕ И АНАЛИЗ ДАННЫХ

НАУЧНЫЙ ЖУРНАЛ

2020 • Том. 10 • № 3

MODELLING AND DATA ANALYSIS

SCIENTIFIC JOURNAL

2020 • Vol. 10 • № 3



Московский государственный
психолого-педагогический университет
Moscow State University
of Psychology & Education

РЕДАКЦИОННАЯ КОЛЛЕГИЯ

Главный редактор – Л.С. Куравский

Заместители главного редактора – С.Д. Кулик, А.В. Пантелеев

Члены редакционной коллегии – К.К. Абгарян, Г.Г. Амосов, М.В. Воронов, Е.Л. Григоренко (США), В.К. Захаров, А.И. Кибзун, Л.М. Либкин (Великобритания), В.Р. Милов, А.В. Наумов, Д.Л. Ревизников, Х. Холлинг (Германия), Д. Фрэнсис (США), К.В. Хорошенко (Великобритания), Г.А. Юрьев

РЕДАКЦИОННЫЙ СОВЕТ

Председатель редакционного совета – Г.Г. Амосов

Члены редакционного совета – В.А. Барабанщиков, П. Бентлер (США), А.В. Горбатов, Л.С. Куравский, Л.М. Либкин (Великобритания), А.А. Марголис, В.В. Рубцов, Д.В. Ушаков, Д. Фрэнсис (США)

Ответственный секретарь – Н.Е. Юрьева

Издаётся с 2011 года

Учредитель

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Московский государственный психолого-педагогический университет»

Адрес редколлегии:

г. Москва, ул. Сретенка, 29, факультет информационных технологий
Тел.: +7 (499) 167-66-74
E-mail: mad.mgppu@gmail.com

Журнал зарегистрирован в Государственном комитете РФ по печати.

Свидетельство о регистрации средств массовой информации

ПИ № ФС77-52058 от 7 декабря 2012 года

ISSN: 2219-3758

ISSN: 2311-9454 (online)

© **ФГБОУ ВО «Московский государственный психолого-педагогический университет», 2020.**
Все права защищены. Любая часть этого издания не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения редакционной коллегии. Правила оформления рукописей, направляемых в редакцию журнала, высылаются по запросу по электронной почте.



How Does a Toddler Experience Digital Media? A Case Study of a 28-Month Old Child Learning to Use the iPad

Marina A. Zhukova*

University of Houston, Houston, TX, USA

Laboratory of Translational Developmental

Sciences, Saint-Petersburg State University, Saint-Petersburg, Russia

ORCID: <https://orcid.org/0000-0002-3069-570X>

e-mail: marina.zhukova@times.uh.edu

Hechmi Kilani**

Genetic and Neurobehavioral Systems: Interdisciplinary Studies,

University of Houston, Houston, TX, USA

ORCID: <https://orcid.org/0000-0002-2424-2165>

e-mail: hechmi.kilani@times.uh.edu

Jessica L. Garcia***

Genetic and Neurobehavioral Systems: Interdisciplinary Studies,

University of Houston, Houston, TX, USA

ORCID: <https://orcid.org/0000-0003-1905-3658>

e-mail: jessica.garcia@times.uh.edu

Mei Tan****

Texas Institute of Measurement, Evaluation and Statistics,

University of Houston, Houston, TX, USA

ORCID: <https://orcid.org/0000-0001-6408-6091>

e-mail: Mei.tan@times.uh.edu

Paul D. Parell*****

Yale University, New Haven, CT, US;

Khan Academy, San Mateo, CA, US

ORCID: <https://orcid.org/0000-0002-6169-5403>

e-mail: pdparell@gmail.com

Sascha Hein*****

Department of Education and Psychology,

Freie Universität Berlin, Berlin, Germany

ORCID: <https://orcid.org/0000-0002-5358-704X>

e-mail: Sascha.Hein@fu-berlin.de



Elena L. Grigorenko*****

Moscow State University of Psychology and Education, Moscow, Russia
Saint-Petersburg State University, Saint-Petersburg, Russia
University of Houston, Houston, TX, USA
Baylor College of Medicine, Houston, TX, USA
Yale University, New Haven, CT, USA
ORCID: <https://orcid.org/0000-0001-9646-4181>
e-mail: elena.grigorenko@times.uh.edu

For citation:

Zhukova M.A., Kilani H., Garcia J.L., Tan M., Parell P.D., Hein S., Grigorenko E.L. How Do Toddlers Experience Digital Media? A Case Study of a 28-Month Old Child Learning to Use iPad. *Modelirovanie i analiz dannykh = Modelling and Data Analysis*, 2020. Vol. 10, no. 3, pp. 5–23. DOI: <https://doi.org/10.17759/mda.2020100301>.

Funding. This project was funded by the US Department of Education under authority of the SCIENCE REFORM ACT OF 2002, Program Title Education Research, DFDA/Subprogram No: 84.305A PR/AWARD NUMBER 5A160402. Grantees undertaking such projects are encouraged to freely express their professional judgement. Therefore, this article does not necessarily reflect the position or policies of the abovementioned agency, and no official endorsement should be inferred.

Acknowledgements. The preparation of this manuscript was supported by funding from the Institutes of Education Sciences, through award R305A160402 (PI: Elena L. Grigorenko). Grantees undertaking such projects are encouraged to express freely their professional judgment. This article, therefore, does not necessarily represent the position or policies of the IES and no official endorsement should be inferred.

***Marina A. Zhukova**, Graduate Research Assistant; University of Houston, Houston, TX, USA; Associate research scientist; Laboratory of Translational Developmental Sciences, Saint-Petersburg State University, Saint-Petersburg, Russia, ORCID: <https://orcid.org/0000-0002-3069-570X>, e-mail: marina.zhukova@times.uh.edu

****Hechm Kilani**, Graduate Research Assistant, Genetic and Neurobehavioral Systems: Interdisciplinary Studies, University of Houston, Houston, TX, USA, ORCID: <https://orcid.org/0000-0002-2424-2165>, e-mail: hechmi.kilani@times.uh.edu

*****Jessica L. Garcia**, Graduate Research Assistant, Genetic and Neurobehavioral Systems: Interdisciplinary Studies, University of Houston, Houston, TX, USA, ORCID: <https://orcid.org/0000-0003-1905-3658>, e-mail: jessica.garcia@times.uh.edu

*****Mei Tan**, Graduate Research Assistant, Texas Institute of Measurement, Evaluation and Statistics, University of Houston, Houston, TX, USA, ORCID: <https://orcid.org/0000-0001-6408-6091>, e-mail: Mei.tan@times.uh.edu

*******Paul D Parell**, Graduate, Yale University, New Haven, CT, US, Product manager, Khan Academy, San Mateo, CA, US, ORCID: <https://orcid.org/0000-0002-6169-5403>, e-mail: pdparell@gmail.com

*******Sascha Hein**, PhD, Professor, Department of Education and Psychology, Freie Universität Berlin, Berlin, Germany, ORCID: <https://orcid.org/0000-0002-5358-704X>, e-mail: Sascha.Hein@fu-berlin.de

*******Elena L. Grigorenko**, PhD, Leading scientist, Moscow State University of Psychology and Education, Moscow, Russia; Principal investigator, Saint-Petersburg State University, Saint-Petersburg, Russia; Professor, University of Houston, Houston, TX, USA; Professor, Baylor College of Medicine, Houston, TX, USA; Professor, Yale University, New Haven, CT, USA, ORCID: <https://orcid.org/0000-0001-9646-4181>, e-mail: elena.grigorenko@times.uh.edu



Digital media is becoming increasingly prevalent in households and classroom settings lowering the age of first exposure to technology; however, little is known about how children experience digital media and how their experiences relate to learning and social engagement. In this case study, we followed a 28-month-old boy in a naturalistic setting for 5 weeks, videotaping the process of his interactions with the iPad. Using a combination of video coding, screen recordings, and a formal academic assessment we evaluated the effects of digital media on specific learning outcomes, child affective states associated with gains in digital literacy, and the role of social interaction in the process of digital media exposure. We found that the number of errors predicted a significant amount of the variance in the child's levels of frustration, attentiveness, help-seeking behaviors, and persistence. Two main types of affect, confusion and attentiveness, predicted gains in his iPad proficiency, providing important insights into the role of emotions in digital learning. With increased levels of iPad proficiency, we observed an inconsistent pattern of child social engagement. This study is the first, to our knowledge, to use a combination of qualitative observation data, standardized assessment, and quantitative analysis of a child's behavior in the process of digital literacy learning.

Keywords: toddler, technology, digital literacy, iPad

INTRODUCTION

Tablet technologies are the most novel addition of digital media to children's environments, as they can be used both as an active and passive technology and require fine motor skills to allow for navigation. The impact of these features on the development of young children currently remains understudied. Compared to the research on tablet use and socializing in adolescence, middle childhood (6 to 12 years old) is considered understudied and toddlerhood (0 to 3 years of age) considered overlooked [1].

The American Academy of Pediatrics recommends that children under the age of 2 should not have any screen time except for video chat [2]. Children between the ages of 2 and 5 years should watch no more than 1 hour of high-quality children's programming per day. However, children under the age of 3 are engaging with this media in various ways alone and with their parents [2]. With today's COVID-19 pandemic, these guidelines might not be as realistic. A recent meta-analysis [3] supported the current pediatric guidelines (mentioned above) on media use by children under the age of 12. The authors analyzed publications from 1960 through March 2019 that included a measure of screen use, a measure of language skills, and statistical criteria that could be transformed into an effect size. Screen use in this context included the use of television, movies, or DVDs on devices such as tablets or television, as well as background television. Forty-two studies (11 of which are from within the past 5 years) accounted for 18,905 participants. A greater quantity of screen use was associated with lower language skills ($r=-0.14$ for screen time, and $r=-0.19$ for background TV), while a better-quality screen use ($r=0.13$ for educational programs, and $r=0.13$ for co-viewing with caregivers) was associated with stronger language skills. A later onset of exposure to digital technology was also associated with stronger language



skills ($r = 0.17$). These findings support that reducing screen exposure, selecting high-quality content, and co-use (how often caregivers joined their children while exposed to screens) of the media are beneficial for children under the age of 12.

Interactions between children and parents when screen-time is involved also plays an important role in children's development. Technology-enhanced storytelling is associated with improved parent-child interaction [4]. To promote learning when using technology and to make a positive difference when learning with digital technologies, adults should be proactive: be present, supportive, and monitor the child's choices, for the technologies to create a positive, learning experience [6]. A study examining the relationship between Internet use and interpersonal communication [7] found that Internet use was significantly correlated ($r = -.13, p < .01$) with decreases in face-to-face family communication, and a decrease in desire for such exchanges ($r = -.120, p < .01$). The study supports the displacement hypothesis by demonstrating that media use, specifically daily Internet use, lowers the time that family spends together and lessens the desire to socialize [7]. In general, it is suggested that screen media, has disruptive effects on the quantity and quality of parent-child interactions. Media use can also be positive, depending on the content of the media used: for example, a longitudinal investigation found that preschool children's viewing of "Sesame Street" suggests a positive effect on children ages 3 to 5 and their vocabulary skills [8]. Other studies suggest that tablet use positively affects fine motor development [9] and promotes visual-spatial skills [10]. Learning from media is associated with cognitive and academic enhancement when programs are age-appropriate and guided by curriculums. Parents are able to maximize positive effects and minimize negative effects of media use by carefully selecting the material to be educational and avoiding purely entertainment and violence focused material. [11]. Learning from media use relies on cognitive demands of a task and a child's cognitive resources which encourages the use of apps that are able to interactively guide attention to important information. For more challenging information, parents are encouraged to use noninteractive video demonstrations for a short term [12].

Digital media is becoming increasingly prevalent in the households and classroom settings lowering the age of first exposure to technology; however, little is known about how children experience digital media and how their experiences relate to learning and social engagement. This case study aims to explore the processes a child goes through to learn to use an iPad, having been introduced to it for the first time at the age of 2 years and 4 months old. We investigate the effects of digital media on specific learning outcomes, child affects associated with gains in digital literacy which refers to proficiency in manipulating and navigating a digital device, and the role of social interaction in the process of digital media exposure.

We hypothesized that the child's performance on an iPad application would increase over time, as well as his digital literacy. We also hypothesized that frustration and confusion would predict the frequency of help-seeking behaviors. In addition, we expected to see a decrease in social interactions and an increase in digital proficiency over time as the child would rely less on a caregiver in problem solving. The findings of Craig and colleagues [13] showed evidence that the level of confusion is critical for optimal learning. Guided by these



results, we hypothesized that confusion and attentiveness would predict child's positive learning outcomes and explain gains in digital literacy and academic knowledge.

METHOD

Participant/Sample

The case subject, Ryan, is a Caucasian male who was 28 months (2 years and 4 months) old at the time of his first iPad exposure. Ryan is raised in a bilingual Russian-English household. He began attending English-based preschool since the age of 18 months. The data presented here were collected while Ryan was with a childcare provider (hereafter, babysitter) outside of his home; he visited the babysitter's house for 3 to 4 hours each time 2 to 3 times per week during the course of the study.

Study Design

During the course of five weeks, Ryan was given the iPad to play with 3 times each week. The researcher and a babysitter sought to assure that the device held his attention for at least 20 minutes per session. In each session, Ryan's interaction with the iPad was videotaped and the iPad's screen recordings were acquired to facilitate formal behavioral coding. Ryan was not given any specific instructions as the focus of the study was on free exploration of digital media. Following Ryan's exposure to the iPad the video recordings of his sessions were coded for a variety of behavioral and performance factors. Further, Ryan's behavior was analyzed to gauge his affect, behavioral characteristics, caregiver interactions during the iPad interactions, and types of manipulations. During the iPad sessions the child was accompanied by a babysitter and/or a researcher (depending on the selected day of recording), who were not given specific instructions and interacted with the child the way they normally would.

Measures

Formal assessments were used to acquire a holistic profile of Ryan's developmental background and academic readiness: The Mullen Scales of Early Learning, MSEL [14], The Clinical Evaluation of Language Fundamentals—4th edition, CELF-4 [15], The Infant-Toddler Social and Emotional Assessment, ITSEA [16], The Conner's Early Childhood Form [17] and the Bracken School Readiness Assessment - Third Edition, BSRA-3 [18]. The school readiness scale was administered three times: to acquire a baseline for Ryan's development prior to iPad exposure, after two weeks of iPad interactions and finally at posttest one week after the iPad sessions discontinued. All assessments were conducted in English.

iPad Application

While the child explored a few applications, for the purpose of the analyses we focused on the Doodle Dots application, as it was the application consistently played throughout the sessions for at least three minutes per sitting. Doodle Dots is a free iPad application released by developer PBS KIDS Sprout in August of 2011. Within the app, children respond to verbal prompts from the game's narrator in order to connect dots and complete a picture. If the wrong dot is selected, the child receives negative feedback (e.g. «That's not a green dot!»), but



there is no limit to the number of incorrect responses allowed or the amount of time needed to connect the dots. The game can be played in 4 modes: numbers, colors, shapes, and fruit.

Video Coding

The video coding system was developed to address Ryan's affect, behavioral characteristics, and caregiver-child interactions during iPad play, and the types of iPad manipulations used. Videos were selected based on their length with a minimum duration of 3 minutes and the length of a coding sequence was based on a single doodle completion. The frequencies of Ryan's behaviors and affects were coded individually, allowing for overlaps. All coding rubrics are summarized in the appendix. The Specific Affect Coding System was used for descriptions of affect facial changes [19]. For the caregiver's behavior, the coding was based on selected categories from the Dyadic Parent-Child Interaction Coding System [20]. Three additional categories (pointing, problem solving, and redirection) were added to the rubric to reflect the specific facilitation of the child's iPad interactions. To analyze the child proficiency on the iPad use, we utilized a 5-point Likert scale developed by J.P. Hourcade, S.L. Mascher, D. Wu, L. Pantoja [21] to rate the ability to operate the device from low (1 point) to very high (5).

RESULTS

Correlations and regressions were completed to test our hypotheses and demonstrate the relative impact of variables on each other. Required assumptions were tested and met for each analysis.

Inter-Rater Reliability

There were five videos recorded throughout the observation period with a mean duration of 6.32 minutes ($SD = 4.14$; $min = 3.57$ minutes, $max = 13.38$ minutes). Coding was carried out by two independent graduate student coders who double-coded two out of nine videos. Interrater reliability was computed as an intraclass correlation coefficient (ICC) for each category of the analysis. For child affect and behavior, an ICC of .75 with a 95% CI [.54 to .89], $F(16,596) = 3.99$, $p < .001$ was achieved. For iPad manipulations the average ICC was .69 with a 95% CI [.41 to .87], $F(16,144) = 3.201$, $p < .001$. The average ICC for the caregiver rubric was .76 with a 95% CI [.57 to .89], $F(16,432) = 4.227$, $p < .001$. For iPad navigation proficiency, the average ICC was .86 with a 95% CI [.62 to .95], $F(16,16) = 4.227$, $p < .001$.

Skill Development

Assessments suggested that Ryan was typically developed in the areas of visual reception and social-emotional development. In language development, Ryan's expressive language was in the Low Average range of ability, which may be due to his bilingual status.

Because Ryan's age fell out of the norming range for the Bracken School Readiness Assessment (BSRA-3), composite scores and percentiles could not be computed and, thus, mastery percentage was used as an indicator of progress. A graphical representation of his mastery across the three time points can be viewed in Figure 1. Ryan's performance distinctly improved in all but one of the measured categories from T1 to T2. His scores then



dropped somewhat from the intermediate assessment to the final assessment, though his performance was still higher than what was observed at the baseline assessment. Given the lack of additional data points the statistical significance of these fluctuations were not studied, and thus these results should be viewed with an understanding that these fluctuations may represent normal variations in skill development typical with this age range.

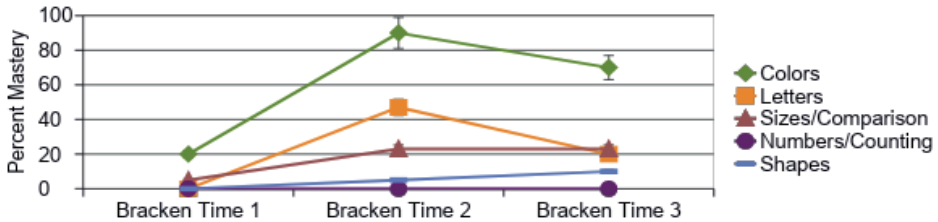


Figure 1. Changes in percent mastery across three administrations of the Bracken School Readiness Assessment, Third Edition (BSRA-3)

Errors

It was initially hypothesized that Ryan would perform better with the game over time. His performance on the game and learning over time was measured on two metrics: academic performance assessed in the number of errors made, and navigation proficiency. For each doodle completed, Ryan received a score from 0 to 5 on how adept he was at navigating the app and following the instructions, which served as his navigation proficiency index. Contrary to the hypothesis, a regression analysis revealed that the number of errors made in the game did not appear to decrease significantly over time, $R^2 = .07$, $F(1,48) = 3.74$, $p = ns$. Time point appeared to be a significant predictor of Ryan's ability to proficiently navigate the application ($R^2 = .11$, $F(1,53) = 6.32$, $p = .01$), though the negative correlation between the two variables ($r = -.33$) indicates that as time progressed, Ryan became less proficient in his use of the app, demonstrating lower accuracy of manipulations.

Affect and Behavior

Aimed at getting a snapshot of the affects and behaviors that a normally developing child experiences when first introduced to a tablet, Ryan's affects and general behaviors were coded. A depiction of the relative occurrence for each affect state, and how Ryan's affect changed over time can be viewed in Figures 2 and 3, respectively. When he was first introduced to Doodle Dots, Ryan would alternate between frustration and confusion. Ryan then reverted to attention seeking behaviors via verbalizations, eye contact, or pointing at the source of the issue. Over time, Ryan's help-seeking lessened in frequency as his frustration reduced.

As shown in Figure 4, eye contact and smiling were the most frequent behaviors that Ryan demonstrated throughout the sessions, both of which typically co-occurred with delight. However, Ryan was also noted to seek eye contact in instances of confusion. Persistence was prominent during the times Ryan got bored and started losing interest. We expected to observe a reduced need for assistance as Ryan's navigation proficiency increased. Interestingly, Ryan sought help regardless of his proficiency level. As can be



viewed in Figure 5, several of the behaviors were noted to decrease over time, particularly those of a social nature.

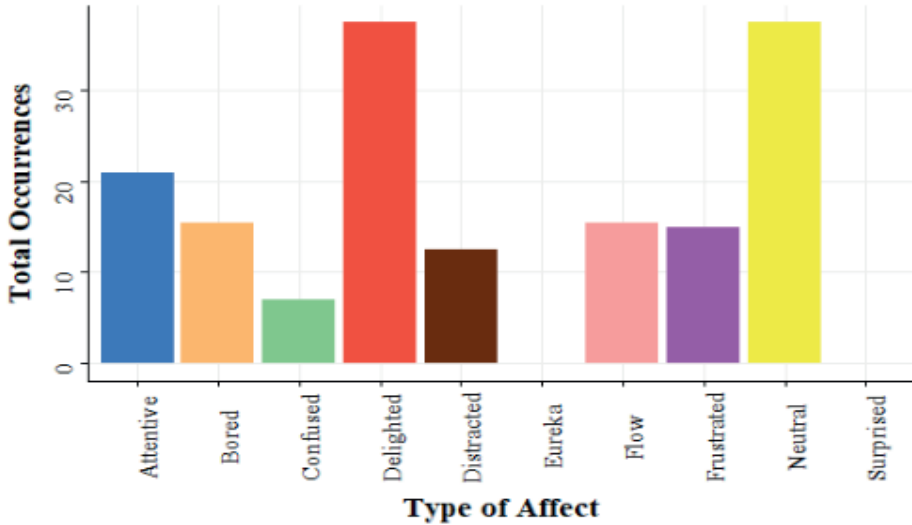


Figure 2. Total occurrences of coded affect states across all time points

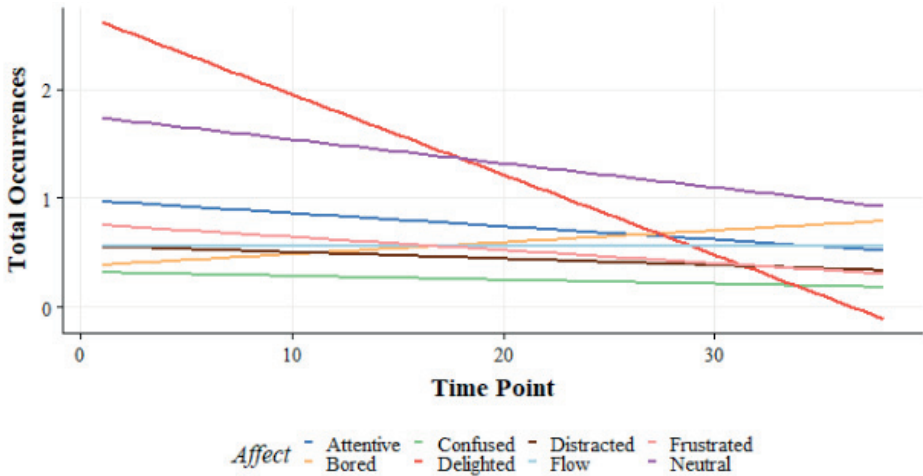


Figure 3. Changes in affect states over time estimated using linear regression. Two types of affect ("eureka" and "surprised") were not included given their lack of occurrence

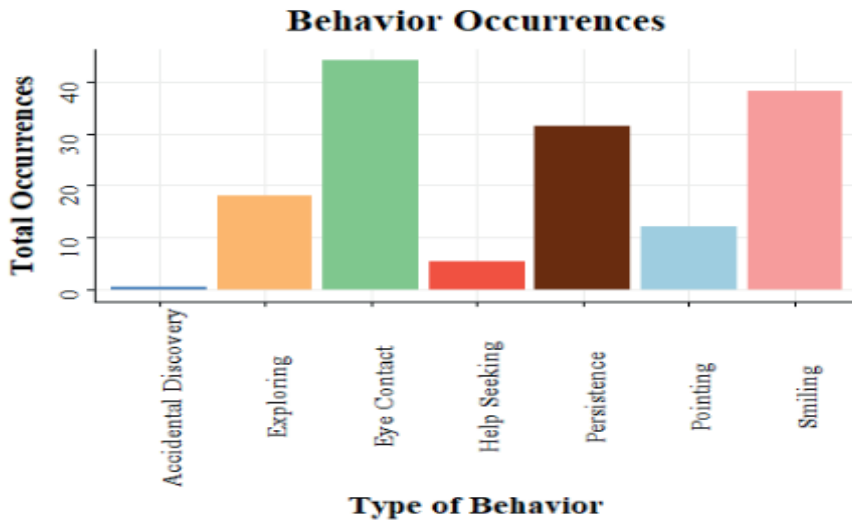


Figure 4. Total occurrences of coded child behaviors across all time points

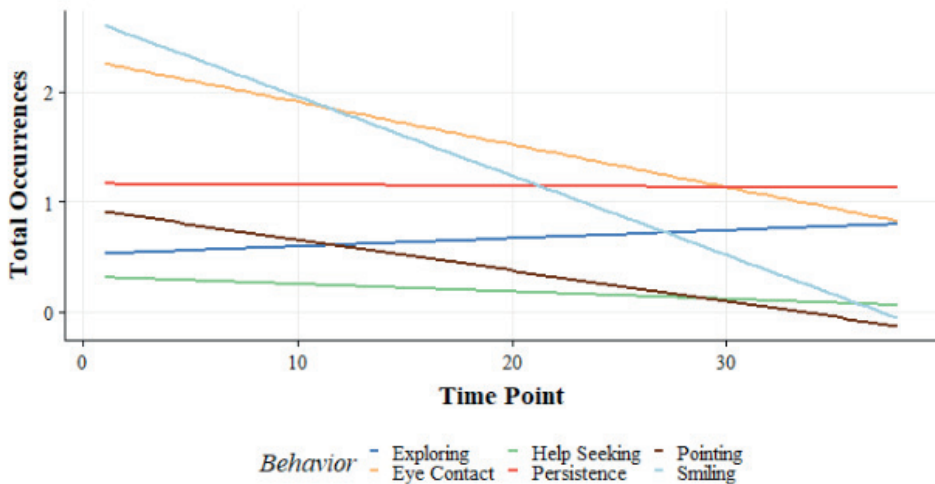


Figure 5. Changes in child behaviors over time estimated using linear regression. One type of behavior (“accidental discovery”) was not included given its infrequency of occurrence

Digital Literacy

The appropriateness and efficiency of iPad manipulations were used as the main metrics to characterize overall index of child’s digital literacy. We conceptualized the appropriateness of manipulations as all manipulations that can be applied to a tablet and lead to meaningful outcomes. All iPad manipulations were mastered by Ryan either independently or through



the observation of the caregiver. Most of Ryan's actions were purposeful and reflected a good understanding of the required actions from the early stages of his iPad interactions. His task performance indicated that some of the concepts were more difficult than others, with shapes producing on average more errors after controlling for the number of times played.

Digital literacy metrics suggested that Ryan started out showing moderate skill in navigating the iPad, and speed and precision of his manipulations increased over time. Some of the inappropriate and ineffective manipulations that were observed included hitting the iPad with a palm or fist, which was typically associated with frustration and boredom. Initially, we hypothesized that Ryan's proficiency in iPad manipulations would increase over time. However, contrary to our prediction, the frequency of inappropriate manipulations increased. One potential explanation could be that Ryan explored iPad functionality by trying out a new behavior repertoire. Another explanation is increased boredom. Over time, Ryan appeared to grow tired of the familiar app and expressed his boredom and frustration in inappropriate manipulations (e.g., sloppiness). In line with this hypothesis is the observation that Ryan began more frequently switching categories in Doodle Dots app (e.g., from colors to fruit) as time went on and was browsing for desired doodles for periods of time longer than that spent completing the actual tasks.

Relation between affect, behavior, and digital performance

Correlational analysis revealed that the number of errors Ryan made on Doodle dots were significantly related to the frequency of affect states, including frustration ($r = .32$, $p = .02$), attentiveness ($r = .47$, $p < .001$), help-seeking ($r = .27$, $p = .05$), and persistence ($r = .42$, $p < .001$). All correlations were positive, which indicates that as errors increased, so did the measured affects and behaviors. It was hypothesized that two types of affect, confusion and attentiveness, would predict navigation proficiency. A multiple regression confirmed this hypothesis, $R^2 = .48$, $F(3,51) = 15.75$, $p < .01$. It was noted that both confusion ($\beta = .40$, $p < .01$) and attentiveness ($\beta = .17$, $p = .01$) significantly predicted variance in navigation proficiency, whereas the interaction between the two affects did not ($\beta = .12$, $p = ns$). Finally, it was predicted that child help seeking would predict navigation proficiency, though analyses revealed this association to be non-significant.

Relation between affect and social behaviors

We hypothesized that social interactions with the caregiver would decrease over time. The results of the regression analysis generally supported this hypothesis. Specifically, it was found that time, as measured by each progressive doodle completed in the game, predicted a significant level of the variance in pointing ($R^2 = .20$, $F(3,51) = 13.51$, $p < .001$), smiling ($R^2 = .46$, $F(3,51) = 46.03$, $p < .001$), and eye contact with the caregiver ($R^2 = .12$, $F(3,51) = 7.42$, $p = .010$). The results of the correlational analyses indicate a moderate negative relation between time and each of these social behaviors ($r = -.45$, $-.68$, $-.35$, respectively) which indicates that these social behaviors decreased over time.

It was also predicted that Ryan's affect, particularly frustration and confusion, would predict frequency of help seeking behaviors. It was found that confusion and the interaction between confusion and frustration predicted a significant amount of the variance in help



seeking behaviors, but frustration alone did not, $R^2 = .21$, $F(3,49) = 4.45$, $p = .01$. We have also found that delight predicted eye contact with the caregiver $R^2 = .34$, $F(1,53) = 26.98$, $p < .001$, $r = .58$.

Caregiver Behaviors

As shown in figure 6, the most frequent form of scaffolding was questions, followed by statements and acknowledgements. Caregiver's questions were primarily reiterations of the Doodle dots instructions ("Where is the blue dot?") or referred to the type of Doodle/category the child was interested in playing next. We observed that the most scaffolding was provided when the child was first introduced to the iPad, with caregiver's interference becoming minimal during the last sessions. Verbal praise was used to encourage the child's exploration and participation. Overall, the frequency breakdown of caregiver's behavior suggests minimal direct involvement (pointing, redirection, problem solving) and caregiver's preference to utilize indirect commands, verbal praises and statements as opposed to negative and direct commands with explicit instructions.

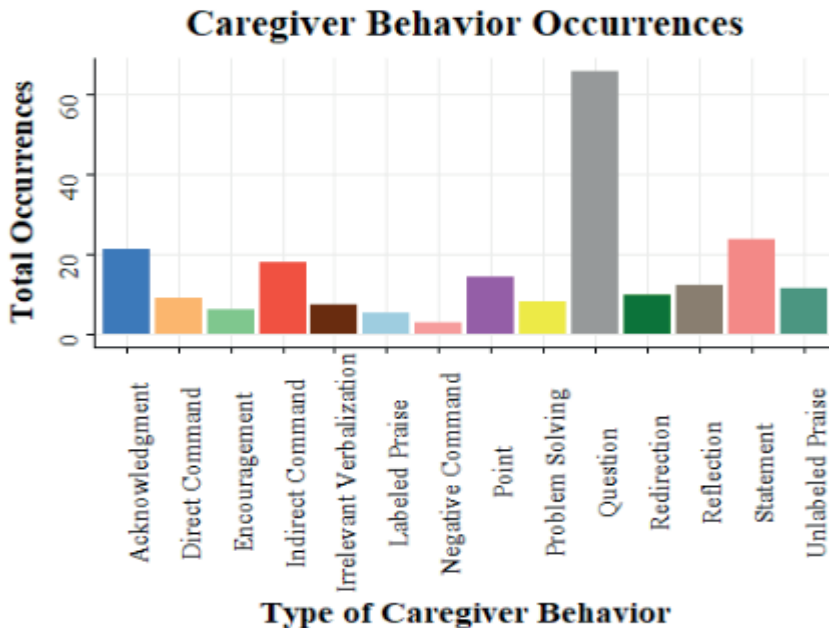


Figure 6. Total occurrences of coded caregiver behaviors across all time points

DISCUSSION

The current study utilized a set of single-case observations of a 28-month-old boy in the naturalistic setting over the course of 5 weeks in the process of his iPad interactions. This



study is the first, to our knowledge, to use a combination of qualitative observation data, standardized assessments and quantitative analysis of child's behavior in the process of digital literacy learning.

Results showed that Ryan's performance on the BSRA-3 distinctly improved from the pre-assessment to the intermediate assessment in the categories of colors, letters and sizes but the categories of numbers and shapes did not show significant improvement. His scores then dropped somewhat from the intermediate assessment to the final assessment, which we believe may be explained by the regression toward the mean. The increase in Ryan's performance on colors, numbers and shapes shaped our hypothesis of knowledge transference as we believed that the increases were due to the iPad application. When we analyzed the number of errors in Doodle Dots over time, results showed that there was no significant decrease in errors. These findings are most likely explained by the fact that Ryan grew bored with the repetitiveness of the task and committed more errors to skip the trial and get to the next Doodle.

A number of studies have shown that affect is not a byproduct of learning but an important factor shaping achievement [e.g., 22]. It has been shown that higher enjoyment and lower boredom predicts greater math achievement in 5th and 6th graders [23]. Another study found a negative association between boredom and learning gains [13]. Interestingly, other authors argue that boredom can be seen as an affect that leads to creativity and exploration [24].

Our findings on digital literacy acquisition show that contrary to our initial hypothesis, the number of functionally inappropriate manipulations did not decrease over time. We also observed an inverse effect with navigation proficiency, indicating that Ryan became less proficient with digital technology over time. We believe that these findings can be explained by Ryan's high persistence, and growing boredom and frustration as he was navigating the app.

Our findings show that during iPad interactions, the leading types of affects were delight and neutral, followed by attentiveness, boredom, flow and frustration. We have noticed that when flow co-occurred with attentiveness, Ryan was the most proficient with his manipulations and concept identification. These observations are corroborated by the findings of S. Craig, A. Graesser, J. Sullins, B. Gholson [13] showing that flow has a meaningful effect size on learning (Cohen's $d = .22$). A few papers have argued that confusion is a corner stone of productive learning as it has been linked to flow and learning gains. In line with these findings, our data showed that confusion and attentiveness (but not their interaction) predicted higher navigation proficiency. However, given the small number of instances that Ryan experienced confusion in the current study, our results should be interpreted with caution. Importantly, we noticed that when confusion was coupled with attentiveness Ryan was more likely to achieve the desired result in his iPad manipulations compared to instances, where confusion was linked to frustration or boredom. This observation is in line with the B. Kort, R. Reilly, R.W. Picard [25] model of constructive learning that shows confusion as an emotion driving positive learning, but frustration as a negative affect hindering learning processes.

The child's behavior involved several instances of help-seeking, which was not reduced as a function of digital proficiency. At the same time, we observed a decrease in other social behaviors over time. These findings point to the differential relationship between types of emotions and sharing. Our study shows that as positive affects become less socially



oriented over time, the child tends to rely on the caregiver only in instances of negative affect. These findings provide an important view on the role of caregiver as a mediator in digital literacy acquisition. We observed that the most frequent forms of scaffolding were questions, followed by statements and acknowledgements. Generally, indirect instructional approaches relying on praise, indirect commands and questions have been shown to be effective in boosting child's exploration, creativity and building self-esteem [26, 27]. We believe that in situations of carefully selected content, low caregiver interference is beneficial for digital literacy learning as it promotes children's creativity and exploration; however, our data shows that boredom and frustration are negatively associated with learning gains and in these instances caregiver's interference would be warranted.

The role of digital media in the lives of toddlers is currently being re-evaluated as families and educational institutions are facing new challenges with COVID-19. Even though the findings on the impact of digital media are controversial, experts agree that balance and moderation is the key to a positive experience with digital media and encourage parents to think more about the quality of the content and less about the quantity [28]. The E-AIMS Model which refers to the media that is that is Engaging, Actively Involved, Meaningful, and Social [29] encourages parents to choose content with clear story lines that children can understand and relate to, and to actively involve the child in the discussion about the content. This idea is also reflected in the guidelines suggesting that the child is developmentally ready for digital content as soon as they are able to talk about their experiences [30]. In our study we observed that Ryan was able to capitalize on his vocabulary to do so as the content corresponded to his current level of verbal skills (i.e., when the doodle was named after completion Ryan was able to repeat the word). At the same time the Doodle Dots application used in the study only partially aligns with the E-AIMS model as the application does not encourage social interaction. Monitoring for the difficulty of the content is an important aspect. The E-AIMS Model encourages parents to promote active involvement of a child instead of being in "autopilot" mode. We believe that the content selected in line with E-AIMS recommendations will allow for some flow experience, where after the initial state of confusion the child effortlessly completes a series of tasks quickly and effortlessly. The flow state promotes competence and has been positively linked to learning gains [31]. In our study we identified that when flow was co-occurring with attentiveness, Ryan was the most proficient with his manipulations and concept identification. We believe that when flow is coupled with a high attention allocation it should be considered an indicator of successful learning rather than autopilot mode. When the activity gets too familiar the child might transition into boredom and a more challenging task should be offered, reflecting the principle of the zone of proximal development. Our findings show that digital literacy acquisition is not linear.

The study has limitations. As an observational case study, external validity is limited and, caution should be taken in generalizing the results of these exploratory analyses until these findings are replicated. In addition, due to the ways in which the videos were coded, the temporal sequence of affects, behaviors, and app-related was not recorded and thus the researchers could not make any inferences about the directionality or causality between the



variables of interest. Future studies should employ the use of advanced software, which would allow to track the sequencing of these variables in order to make stronger inferences about the nature of their relationships. Another limitation is the lack of other educational applications that could have allowed the researchers to explore the extent to which these patterns can be found when the child is exploring other applications. Given that the child in the current study was not directed to use specific applications by the caregiver and was given free rein to explore, it may also be beneficial for future research to examine what drives children to choose specific applications, and subsequent learning experiences, over others.

References

1. Blumberg F.C., Deater-Deckard K., Calvert S.L., Flynn R.M., Green C.S., Arnold D., Brooks P.J. Digital Games as a Context for Children's Cognitive Development: Research Recommendations and Policy Considerations // *Social Policy Report*. – 2019. Vol. 32, № 1. – pp. 1–33.
2. Shifrin D., Brown A., Hill D., Jana L., Flinn S.K. Growing up digital: Media research symposium // *American Academy of Pediatrics*. – 2015. Vol. 1. – pp. 2015.
3. Madigan S., McArthur B.A., Anhorn C., Eirich R., Christakis D.A. Associations Between Screen Use and Child Language Skills: A Systematic Review and Meta-analysis // *JAMA pediatrics*. – 2020.
4. Teepe R., Molenaar I., Verhoeven L. Technology-enhanced storytelling stimulating parent-child interaction and preschool children's vocabulary knowledge // *Journal of Computer Assisted Learning*. – 2017. Vol. 33, № 2. – pp. 123–136.
5. Lauricella A.R., Wartella E., Rideout V.J. Young children's screen time: The complex role of parent and child factors // *Journal of Applied Developmental Psychology*. – 2015. Vol. 36. – pp. 11–17.
6. Stephen C., Plowman L. Digital technologies, play and learning // *Early Childhood Folio*. – 2013. Vol. 17, № 2. – pp. 3–8.
7. Shim Y.S. The Impact of the Internet on Teenagers' Face-to-Face Communication // *Global media journal*. – 2007. Vol. 6, № 10.
8. Rice M.L., Huston A.C., Truglio R., Wright J.C. Words from "Sesame Street": Learning vocabulary while viewing // *Developmental psychology*. – 1990. Vol. 26, № 3. – pp. 421.
9. Bedford R., Saez de Urabain I.R., Cheung C.H., Karmiloff-Smith A., Smith T.J. Toddlers' fine motor milestone achievement is associated with early touchscreen scrolling // *Frontiers in psychology*. – 2016. Vol. 7. – pp. 1108.
10. Nebel S., Schneider S., Rey G.D. Mining learning and crafting scientific experiments: a literature review on the use of minecraft in education and research // *Journal of Educational Technology & Society*. – 2016. Vol. 19, № 2. – pp. 355–366.
11. Kirkorian H.L., Wartella E.A., Anderson D.R. Media and young children's learning // *The Future of children*. – 2008. – pp. 39–61.
12. Kirkorian H.L. When and how do interactive digital media help children connect what they see on and off the screen? // *Child Development Perspectives*. – 2018. Vol. 12, № 3. – pp. 210–214.
13. Craig S., Graesser A., Sullins J., Gholson B. Affect and learning: an exploratory look into the role of affect in learning with AutoTutor // *Journal of educational media*. – 2004. Vol. 29, № 3. – pp. 241–250.
14. Mullen scales of early learning. / Mullen E.M.: AGS Circle Pines, MN, 1995.
15. CELF 4. / Semel E.M., Wiig E.H., Secord W.: PsychCorp, 2004.
16. Carter A.S., Briggs-Gowan M.J. ITSEA: Infant-toddler social and emotional assessment examiner's manual // San Antonio, TX: PsychCorp. – 2006.
17. Conners C.K. Conners third edition (Conners 3) // Los Angeles, CA: Western Psychological Services. – 2008.



18. Bracken school readiness assessment. / Bracken B.A.: Psychological Corporation, 2007.
19. Coan J.A., Gottman J.M. The specific affect coding system (SPAFF) // Handbook of emotion elicitation and assessment. – 2007. – pp. 267–285.
20. Dyadic parent-child interaction coding system: A manual. / Eyberg S.M., Robinson E.A.: American Psychological Association, 1981.
21. Hourcade J.P., Mascher S.L., Wu D., Pantoja L. Look, my baby is using an iPad! An analysis of YouTube videos of infants and toddlers using tablets // Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, – 2015. – pp. 1915–1924.
22. Kytälä M., Björn P.M. Prior mathematics achievement, cognitive appraisals and anxiety as predictors of Finnish students' later mathematics performance and career orientation // Educational Psychology. – 2010. Vol. 30, № 4. – pp. 431–448.
23. Putwain D.W., Becker S., Symes W., Pekrun R. Reciprocal relations between students' academic enjoyment, boredom, and achievement over time // Learning and Instruction. – 2018. Vol. 54. – pp. 73–81.
24. Belton T., Priyadharshini E. Boredom and schooling: a cross-disciplinary exploration // Cambridge Journal of Education. – 2007. Vol. 37, № 4. – pp. 579–595.
25. Kort B., Reilly R., Picard R.W. An affective model of interplay between emotions and learning: Reengineering educational pedagogy-building a learning companion // Proceedings IEEE International Conference on Advanced Learning Technologies –IEEE, 2001. – pp. 43–46.
26. Bonnette R., McBride R.E., Tolson H. The differential effect of indirect instruction in the teaching of sport skills on critical thinking and self-esteem of early adolescent boys placed at risk // Sport, Education and Society. – 2001. Vol. 6, № 2. – pp. 183–198.
27. Ferkany M. The educational importance of self-esteem // Journal of Philosophy of Education. – 2008. Vol. 42, № 1. – pp. 119–132.
28. Media moms & digital dads: A fact-not-fear approach to parenting in the digital age. / Uhls Y.T.: Routledge, 2016.
29. Zero to Three Association. Choosing Media Content for Young Children Using the E-AIMS Model. – 2018. – URL: <https://zero-to-three.s3.amazonaws.com/images/2837/db2be46d-d6e1-42a8-bed1-83ccef9e7208-original.jpg?1540487653> (дата обращения: 07.22.2020).
30. Radesky J., Christakis D., Hill D. Council on Communications and Media // Media and young minds. Pediatrics. – 2016. Vol. 138 (5) e20162591; DOI: <https://doi.org/10.1542/peds.2016-2591>
31. Konradt U., Filip R., Hoffmann S. Flow experience and positive affect during hypermedia learning // British journal of educational technology. – 2003. Vol. 34, № 3. – pp. 309–327.



Как ребенок осваивает электронные технологии? Анализ случая 28-месячного ребенка в процессе освоения iPad

Марина А. Жукова*

Хьюстонский университет, Хьюстон, США
Лаборатория междисциплинарных исследований развития человека
Санкт-Петербургский государственный Университет (Россия)
ORCID: <https://orcid.org/0000-0002-3069-570X>
e-mail: marina.zhukova@times.uh.edu

Хешми Килани**

Лаборатория междисциплинарных исследований генетики и
нейроповеденческих систем; Хьюстонский университет, Хьюстон, США
ORCID: <https://orcid.org/0000-0002-2424-2165>
e-mail: hechmi.kilani@times.uh.edu

Джессика Л. Гарсия***

Лаборатория междисциплинарных исследований генетики и
нейроповеденческих систем; Хьюстонский университет, Хьюстон, США
ORCID: <https://orcid.org/0000-0003-1905-3658>
e-mail: jessica.garcia@times.uh.edu

Мей Тан****

Техасский институт измерений,
оценки и статистики, Хьюстонский университет, США
ORCID: 0000-0001-6408-6091
e-mail: Mei.tan@times.uh.edu

Пол Д. Перэл*****

Йельский университет, Нью Хейвен, Коннектикут, США
Khan Academy, Сан Матео, Калифорния, США
ORCID: <https://orcid.org/0000-0002-6169-5403>
e-mail: pdparell@gmail.com

Саша Хайн*****

Департамент Образования и Психологии,
Свободный Университет Берлина, Германия
ORCID: <https://orcid.org/0000-0002-5358-704X>
e-mail: Sascha.Hein@fu-berlin.de

Елена Л. Григоренко*****

Московский государственный психолого-педагогический
университет, Москва, Российская Федерация
Санкт-Петербургский государственный университет,
Санкт-Петербург, Российская Федерация
Хьюстонский университет, Хьюстон, Техас, США



Бэйлорский медицинский колледж, Хьюстон, Техас, США
Йельский университет, Нью-Хейвен, Коннектикут, США
ORCID: <https://orcid.org/0000-0001-9646-4181>
e-mail: elena.grigorenko@times.uh.edu

Для цитаты:

Жукова М.А., Килани Х., Гарсия Д.Л., Тан М., Перэл П.Д., Хайн С., Григоренко Е.Л.
Как ребенок осваивает электронные технологии? Анализ случая 28-месячного ребенка
в процессе освоения iPad // Моделирование и анализ данных. 2020. Том 10. №3. С. 5-23.
DOI: <https://doi.org/10.17759/mda.2020100301> (In Eng. abstr. in Rus.).

Финансирование. Данный проект был поддержан департаментом образования США SCIENCE REFORM ACT OF 2002, Program Title Education Research, DFDA/Subprogram No: 84.305A PR/AWARD NUMBER 5A160402. Авторы несут ответственность за содержание публикации, которое не обязательно отражает официальную позицию грантодателя.

***Марина А. Жукова**, Кандидат психологических наук, Ассистент-исследователь, Хьюстонский университет (США); Научный сотрудник Лаборатории междисциплинарных исследований развития человека Санкт-Петербургский государственный Университет, Россия, ORCID: <https://orcid.org/0000-0002-3069-570X>, e-mail: marina.zhukova@times.uh.edu

****Хешми Килани**, Ассистент-исследователь Лаборатории междисциплинарных исследований генетики и нейроповеденческих систем; Хьюстонский университет, Хьюстон, США, ORCID: <https://orcid.org/0000-0002-2424-2165>, e-mail: hechmi.kilani@times.uh.edu

*****Джессика Л. Гарсия**, Ассистент-исследователь Лаборатории междисциплинарных исследований генетики и нейроповеденческих систем; Хьюстонский университет, США, ORCID: <https://orcid.org/0000-0003-1905-3658>, e-mail: jessica.garcia@times.uh.edu

******Мей Тан**, Ассистент-исследователь, Техасский институт измерений, оценки и статистики, Хьюстонский университет, Хьюстон, США, ORCID: <https://orcid.org/0000-0001-6408-6091>, e-mail: Mei.tan@times.uh.edu

*******Пол Д. Перэл**, Выпускник Йельского университета, Нью Хейвен, Коннектикут, США Менеджер по продукту, Khan Academy, Сан Матео, Калифорния, США, ORCID: <https://orcid.org/0000-0002-6169-5403>, e-mail: pdparell@gmail.com

*******Саша Хайн**, PhD, Профессор, Департамент Образования и Психологии, Свободный Университет Берлина, Германия, ORCID: <https://orcid.org/0000-0002-5358-704X>, e-mail: Sascha.Hein@fu-berlin.de

*******Елена Л. Григоренко**, PhD, доктор психологических наук, ведущий научный сотрудник, Московский государственный психолого-педагогический университет, Москва, Российская Федерация; главный исследователь, Санкт-Петербургский государственный университет, Санкт-Петербург, Российская Федерация; профессор, Хьюстонский университет, Хьюстон, Техас, США; профессор, Бэйлорский медицинский колледж, Хьюстон, Техас, США; профессор, Йельский университет, Нью-Хейвен, Коннектикут, США, ORCID: <https://orcid.org/0000-0001-9646-4181>, e-mail: elena.grigorenko@times.uh.edu



Электронные технологии набирают все большую популярность и представлены как в домашнем окружении, так и в образовательных учреждениях, снижая возраст первого контакта ребенка с электронными устройствами; при этом существует недостаток исследований, посвященных процессу освоения электронных технологий детьми дошкольного возраста и связи данного процесса с социальным взаимодействием. В данном исследовании мы рассматриваем случай 28-месячного ребенка в процессе обучения взаимодействию с планшетным компьютером. Данные взаимодействия регистрировались на видео в течение 5 недель. Используя комбинацию кодирования видеозаписей, анализа записей экрана и стандартизованного академического тестирования нами были проанализированы эффекты, оказываемые электронными технологиями на различные аспекты обучения, аффективное состояние ребенка, а также роль социального взаимодействия в процессе освоения технологий. Результаты исследования свидетельствуют о том, что количество ошибок предсказывало состояние фрустрации, уровень внимания ребенка, частоту обращения ко взрослому за помощью, а также настойчивость. Два ключевых состояния ребенка, затруднение и внимательность, были связаны с улучшением навыков пользования iPad. Полученные данные вносят важный вклад в понимание роли эмоционального компонента в процессе научения. По мере нарастания навыка пользования iPad, ребенок демонстрировал противоречивый паттерн социального взаимодействия. Данное исследование является первым, использующим комбинацию качественной оценки данных наблюдения, стандартизованного тестирования и количественного анализа поведения ребенка в процессе приобретения технической грамотности.

Ключевые слова: тоддлер, технологии, техническая грамотность, iPad.

Литература

1. Blumberg F.C., Deater-Deckard K., Calvert S.L., Flynn R.M., Green C.S., Arnold D., Brooks P.J. Digital Games as a Context for Children's Cognitive Development: Research Recommendations and Policy Considerations // Social Policy Report. – 2019. – Т. 32, № 1. – С. 1–33.
2. Shifrin D., Brown A., Hill D., Jana L., Flinn S.K. Growing up digital: Media research symposium // American Academy of Pediatrics. – 2015. – Т. 1. – С. 2015.
3. Madigan S., McArthur B.A., Anhorn C., Eirich R., Christakis D.A. Associations Between Screen Use and Child Language Skills: A Systematic Review and Meta-analysis // JAMA pediatrics. – 2020.
4. Teepe R., Molenaar I., Verhoeven L. Technology-enhanced storytelling stimulating parent-child interaction and preschool children's vocabulary knowledge // Journal of Computer Assisted Learning. – 2017. – Т. 33, № 2. – С. 123–136.
5. Lauricella A.R., Wartella E., Rideout V.J. Young children's screen time: The complex role of parent and child factors // Journal of Applied Developmental Psychology. – 2015. – Т. 36. – С. 11–17.
6. Stephen C., Plowman L. Digital technologies, play and learning // Early Childhood Folio. – 2013. – Т. 17, № 2. – С. 3–8.
7. Shim Y.S. The Impact of the Internet on Teenagers' Face-to-Face Communication // Global media journal. – 2007. – Т. 6, № 10.
8. Rice M.L., Huston A.C., Truglio R., Wright J.C. Words from "Sesame Street": Learning vocabulary while viewing // Developmental psychology. – 1990. – Т. 26, № 3. – С. 421.
9. Bedford R., Saez de Urabain I.R., Cheung C.H., Karmiloff-Smith A., Smith T.J. Toddlers' fine motor milestone achievement is associated with early touchscreen scrolling // Frontiers in psychology. – 2016. – Т. 7. – С. 1108.



10. Nebel S., Schneider S., Rey G.D. Mining learning and crafting scientific experiments: a literature review on the use of minecraft in education and research // *Journal of Educational Technology & Society*. – 2016. – T. 19, № 2. – C. 355–366.
11. Kirkorian H.L., Wartella E.A., Anderson D.R. Media and young children's learning // *The Future of children*. – 2008. – C. 39–61.
12. Kirkorian H.L. When and how do interactive digital media help children connect what they see on and off the screen? // *Child Development Perspectives*. – 2018. – T. 12, № 3. – C. 210–214.
13. Craig S., Graesser A., Sullins J., Gholson B. Affect and learning: an exploratory look into the role of affect in learning with AutoTutor // *Journal of educational media*. – 2004. – T. 29, № 3. – C. 241–250.
14. Mullen scales of early learning. / Mullen E.M.: AGS Circle Pines, MN, 1995.
15. CELF 4. / Semel E.M., Wiig E.H., Secord W.: PsychCorp, 2004.
16. Carter A.S., Briggs-Gowan M.J. ITSEA: Infant-toddler social and emotional assessment examiner's manual // San Antonio, TX: PsychCorp. – 2006.
17. Conners C.K. Conners third edition (Conners 3) // Los Angeles, CA: Western Psychological Services. – 2008.
18. Bracken school readiness assessment. / Bracken B.A.: Psychological Corporation, 2007.
19. Coan J.A., Gottman J.M. The specific affect coding system (SPAFF) // *Handbook of emotion elicitation and assessment*. – 2007. – C. 267–285.
20. Dyadic parent-child interaction coding system: A manual. / Eyberg S.M., Robinson E.A.: American Psychological Association, 1981.
21. Hourcade J.P., Mascher S.L., Wu D., Pantoja L. Look, my baby is using an iPad! An analysis of YouTube videos of infants and toddlers using tablets // *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, – 2015. – C. 1915–1924.
22. Kyttälä M., Björn P.M. Prior mathematics achievement, cognitive appraisals and anxiety as predictors of Finnish students' later mathematics performance and career orientation // *Educational Psychology*. – 2010. – T. 30, № 4. – C. 431–448.
23. Putwain D.W., Becker S., Symes W., Pekrun R. Reciprocal relations between students' academic enjoyment, boredom, and achievement over time // *Learning and Instruction*. – 2018. – T. 54. – C. 73–81.
24. Belton T., Priyadarshini E. Boredom and schooling: a cross-disciplinary exploration // *Cambridge Journal of Education*. – 2007. – T. 37, № 4. – C. 579–595.
25. Kort B., Reilly R., Picard R.W. An affective model of interplay between emotions and learning: Reengineering educational pedagogy-building a learning companion // *Proceedings IEEE International Conference on Advanced Learning Technologies –IEEE, 2001*. – C. 43–46.
26. Bonnette R., McBride R.E., Tolson H. The differential effect of indirect instruction in the teaching of sport skills on critical thinking and self-esteem of early adolescent boys placed at risk // *Sport, Education and Society*. – 2001. – T. 6, № 2. – C. 183–198.
27. Ferkany M. The educational importance of self-esteem // *Journal of Philosophy of Education*. – 2008. – T. 42, № 1. – C. 119–132.
28. Media moms & digital dads: A fact-not-fear approach to parenting in the digital age. / Uhls Y.T.: Routledge, 2016.
29. Zero to Three Association. Choosing Media Content for Young Children Using the E-AIMS Model. – 2018. – URL: <https://zero-to-three.s3.amazonaws.com/images/2837/db2be46d-d6e1-42a8-bed1-83ccef9e7208-original.jpg?1540487653> (дата обращения: 07.22.2020).
30. Radesky J., Christakis D., Hill D. Council on Communications and Media // *Media and young minds. Pediatrics*. – 2016. – T. 138 (5) e20162591; DOI: <https://doi.org/10.1542/peds.2016-2591>
31. Konradt U., Filip R., Hoffmann S. Flow experience and positive affect during hypermedia learning // *British journal of educational technology*. – 2003. – T. 34, № 3. – C. 309–327.

Автоматическая кластеризация документов СМИ на основе анализа их смыслового содержания

Кан А.В.*

МАИ, г. Москва, Российская Федерация
ORCID: <https://orcid.org/0000-0001-9410-406X>
e-mail: kan_a@mail.ru

Козловская Я.Д.**

МАИ, г. Москва, Российская Федерация
ORCID: <https://orcid.org/0000-0002-1780-5687>
e-mail: yana_kozlovskaja@mail.ru

Кадушкин Н.А.***

МАИ, г. Москва, Российская Федерация
ORCID: <https://orcid.org/0000-0002-0327-909X>
e-mail: bbamrin@gmail.com

Хорошилов Ал-др.А.****

ГосНИИАС, г. Москва, Российская Федерация
ORCID: <https://orcid.org/0000-0003-4885-3232>
e-mail: aleksandr_khor@mail.ru

Для цитаты:

Кан А.В., Козловская Я.Д., Кадушкин Н.А., Хорошилов Ал-др А. Автоматическая кластеризация документов СМИ на основе анализа их смыслового содержания // Моделирование и анализ данных. 2020. Том 10. № 3. С. 24–38. DOI: <https://doi.org/10.17759/mda.2020100302>

***Кан Анна Владимировна**, доцент Московского авиационного института (национальный исследовательский университет), г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0001-9410-406X>, e-mail: kan_a@mail.ru

****Козловская Яна Дмитриевна**, магистрант Московского авиационного института (национальный исследовательский университет), г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0002-1780-5687>, e-mail: yana_kozlovskaja@mail.ru

*****Кадушкин Николай Алексеевич**, студент Московского авиационного института (национальный исследовательский университет), г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0002-0327-909X>, e-mail: bbamrin@gmail.com

******Хорошилов Александр Александрович**, инженер Государственного научно-исследовательского института авиационных систем (ГосНИИАС), г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0003-4885-3232>, e-mail: aleksandr_khor@mail.ru



В статье описывается решение проблемы автоматической кластеризации документов средств массовой информации (СМИ) на основе их смыслового анализа. Предлагаемое решение базируется на методах машинной грамматики, семантико-синтаксического и концептуального анализа текстов, а также на методах выявления понятийного состава коллекции документов и формализации смыслового содержания текстов. Разработанный алгоритм процесса кластеризации документов обеспечивает возможность его реализации в полностью автоматическом режиме без предварительного машинного обучения.

Ключевые слова: автоматическая кластеризация документов, машинная грамматика, семантико-синтаксический анализ текстов, концептуальный анализ текстов, актуальный концептуальный словарь.

ВВЕДЕНИЕ

Основная идея кластеризации коллекции документов заключается в разделении этой коллекции на группы (кластеры), совпадающих по смысловому содержанию. Это можно достигнуть на основе установления принципов сходства и различия документов. Сходство содержания документов можно установить наличием в них сходной по смыслу системы понятий, а различие определяется отсутствием таких понятий. При решении этой задачи необходимо из всей системы понятий, содержащейся в коллекции документов выделить ту систему значимых (ключевых) понятий, которая обладает достаточной смыслоразличительной способностью, обеспечивающей возможность разделения документов на кластеры с заданной полнотой и точностью. Выявление такой системы понятий можно обеспечить методами концептуального анализа лексического состава коллекции текстов. Такая система понятий должна составить основу семантического словаря, с помощью которого будет выявляться понятийная структура документов и строиться их поисковые образы.

Автоматическое создание словаря понятий по корпусу текстов достаточно подробно изложено в работе [8]. Отметим только то, что в основу концептуального анализа текста положены синтаксические, семантические и статистические методы анализа текстов.

Синтаксические методы позволяют выявить синтаксическую роль значимых слов и словосочетаний в предложении путем установления их синтаксической роли (принадлежность к словосочетаниям, являющимся в предложении группой подлежащего (субъекта), группой сказуемого (предиката) или группой дополнения (объекта)) [6–10].

Семантические методы позволяют выявить значимые в предметной области слова и словосочетания путем их соотнесения с элементами эталонных словарей или их формализованных семантических признаков с эталонными представлениями этих признаков [6–10].

Статистические методы позволяют путем назначения весовых коэффициентов установить состав значимых для коллекции документов понятий, выраженных словами и словосочетаниями, на основе анализа частот их появления в коллекции документов [2].



ТЕОРЕТИЧЕСКОЕ ОБОСНОВАНИЕ МЕТОДА КЛАСТЕРИЗАЦИИ

Предлагаемое решение задачи кластеризации базируется на концепции фразеологического концептуального анализа текстов [7; 8]. В соответствии с этой концепцией смысловое содержание текстов определяется системой текстовых понятий, проф. Г.Г. Белоногов определяет термин «*понятие*» как «*социально значимый мыслительный образ, за которым в языке закреплено его наименование в виде отдельного слова или, значительно чаще, в виде устойчивого фразеологического словосочетания...*» [6], и их смысловых отношений, входящих в состав текстовых предложений. Смысл каждого предложения текста выражается через предикатно-актантную структуру.

Предикатно-актантная структура (ПАС) – описание предложений в виде понятий-актантов, выступающих в роли описываемых объектов, и понятий-предикатов, определяющих связи между объектами.

Предикат (*Predicate*), субъект (*Subject*) и объект (*Object*) представляются в виде шаблонов – обобщенных синтагм в нормализованном виде:

$$P = \{GS_{norm_x} \mid x \in X\}, \quad (1)$$

$$S = \{GS_{norm_y} \mid y \in Y\}, \quad (2)$$

$$O = \{GS_{norm_z} \mid z \in Z\}, \quad (3)$$

где x, y, z – нормированный набор индексов шаблона из словаря D обобщенных синтагм;

$X, Y, Z \in D$ – множества синтагм соответствующих элементов ПАС из словаря D ;

GS_{norm} – обобщенная синтагма в нормализованном виде (лемма):

$$GS_{norm} = Lemma = \{GS \mid n = 1, c = 1\}, \quad (4)$$

где GS – обобщенная синтагма, полученная при анализе всех допустимых грамматических признаков элементов ПАС предложения или словосочетания.

Смысл каждого предложения текста $SSen$ представляется через его предикатно-актантную структуру:

$$SSen = \sum_{i,j,k=1}^{I,J,K} \{S_i, P_j, O_k\} + \sum_{q=0}^Q R_q, \forall R \in \mathbb{R}, \quad (5)$$

где I, J, K, Q – счетные множества.

Идея кластеризации текстов СМИ по смысловому содержанию базируется на следующих постулатах и решениях:

1. Содержание каждого документа описывается системой значимых в предметной области понятий.
2. Основное смысловое содержание документа определяется системой его ключевых (значимых для предметной области) понятий.



3. Значимость понятий (веса смысловой значимости в предметной области) определяется статистическими, синтаксическими и семантическими критериями.
4. Понятия могут иметь различное текстовое представление, но при этом быть близким по своему смысловому содержанию.
5. Понятия в тексте связаны как парадигматическими (внеконтекстными), так и синтагматическими (контекстными) отношениями.
6. Синтагматические отношения разрешаются на основе семантико-синтаксического анализа текстов.
7. Парадигматические отношения разрешаются на основе использования таких семантических инструментов, как тезаурусы и онтологии.
8. Установление смыслового тождества понятий, представленных одним и тем же лексическим составом в различном синтаксическом контексте окружения, разрешаются путем нормализации их лексического состава на уровне словоизменения или словообразования.

Общее решение задачи кластеризации обеспечивается следующими технологическими операциями.

1. Создание актуального словаря по текстам коллекции документов, полученного автоматически из частотного словаря путем исключения из него высокочастотной и низкочастотной частей этого словаря.
2. Повышения распознающей способности семантически значимой лексики в актуальном словаре, путем их сопоставления с наиболее значимыми понятиями предметной области.
3. Генерации на основе актуального словаря понятийных образов документов (ПОД), отражающих основное смысловое содержание документа.
4. Формирования кластеров путем последовательного сопоставления ПОДов документов-эталонов и ПОДов всех документов коллекции, и в случае превышения пороговых значений такие документы входят в состав кластера. После окончания каждого процесса формирования кластера этот документ-эталон и документы его кластера исключаются из «черного списка».
5. Процесс кластеризации завершается в случае полного опустошения «черного списка».
6. На завершающем этапе производится установление документа-центроида кластера (документа, смысловое содержание которого в наибольшей степени отражаются обобщенное содержание документов кластера).

РЕШЕНИЕ ЗАДАЧИ КЛАСТЕРИЗАЦИИ КОЛЛЕКЦИИ ДОКУМЕНТОВ

Рение задачи кластеризации можно условно разделить на несколько этапов, в рамках которых производятся технологические операции формализации представления содержанием отдельных документов, выявления смыслоразличающего понятийного состава коллекции документов и представления обобщенного содержания документов кластеров в виде дайджеста кратких рефератов.



На первом этапе нужно определить множество слов и словосоч. етаний признакового пространства, по которым будет производиться установление смысловой схожести содержания документов коллекции. Для этого необходимо определить: по каким критериям нужно оценивать смысловую значимость каждого элемента признакового пространства и в соответствии с этими критериями назначить каждому слову или словосочетанию весовой коэффициент их смысловой значимости в признаковом пространстве.

В качестве меры смысловой значимости слов и словосочетаний было решено использовать статистическую меру $TF - IDF$ (TF (term frequency) – частота слова).

TF – отношение числа вхождений наименования понятия общему числу наименований понятий документа. Таким образом, оценивается важность наименования понятия t_i в пределах отдельного документа.

$$TF(t, d) = \frac{n_t}{\sum_k n_k}, \quad (6)$$

где n_t – число вхождений наименования понятия t в документ;

$\sum_k n_k$ – общее число наименований понятий в данном документе.

IDF – инверсия частоты, с которой некоторое наименование понятия встречается в документах коллекции. Для каждого уникального слова в пределах конкретной коллекции документов существует только одно значение IDF :

$$IDF(t, D) = \log \frac{|D|}{|\{d_i \in D \mid t \in d_i\}|}, \quad (7)$$

где $|D|$ – число документов в коллекции; $|\{d_i \in D \mid t \in d_i\}|$ – число документов из коллекции D , в которых встречается t (когда $n_t \neq 0$).

Мера $TF - IDF$ является произведением двух сомножителей:

$$TF - IDF(t, d, D) = TF(t, d) \times IDF(t, D). \quad (8)$$

Большой вес в $TF - IDF$ получат наименования понятий с высокой частотой встречаемости в конкретном документе и с относительно низкой частотой в пределах всего корпуса текстов.

Но данная статистическая мера в явном виде не отражает смысловую составляющую наименований понятий. С этой целью была разработана система коррелирующих семантических весовых коэффициентов наименований понятий, восполняющих этот пробел. Обобщенная мера смысловой значимости наименований понятий M_7 с учетом этих коэффициентов вычисляются по формуле

$$M_7 = (TF \cdot IDF) \cdot K_n \cdot K_z \cdot K_t \cdot K_w \cdot K_s \cdot K_f, \quad (9)$$

где:

K_n – коэффициент, учитывающий распознающую способность слов при их нормализации.

K_z – коэффициент, учитывающий вхождение в заголовки слов или словосочетаний.

K_t – коэффициент, учитывающий вхождение в тезаурус слов или словосочетаний.

K_w – коэффициент, учитывающий количество слов в словосочетании.



K_w – коэффициент, учитывающий синтаксическую роль слова или словосочетания в предложении.

K_f – коэффициент, учитывающий принадлежность понятия к фамильно-именной группе, бренду и др.

Для реализации процесса установления меры смысловой значимости наименований понятий коллекции документов было выполнено исследование по созданию актуального семантического словаря (АСК) для кластеризации текстов методами приближенного концептуального анализа текстов [6]. В рамках этого исследования был составлен частотный словарь наименований понятий и была сформирована его характеристическая таблица, устанавливающая закономерности покрытия текстов понятиями, содержащимися в коллекции документов, и рангами значений их частот. В табл. 1. представлена характеристическая таблица частотного словаря наименований понятий коллекции документов общим объемом 2013 документов.

Таблица 1

**Характеристическая таблица частотного словаря
наименований понятий коллекции документов**

Ранг частот R_f	Частота f	Кратность m	Накопленная частота A_f	Накопленная кратность A_m	Относительная накопл. част. C
1	55255	1	55255	1	0,030403
2	42487	1	97742	2	0,053780
3	38668	1	136410	3	0,075056
4	38112	1	174522	4	0,096026
5	20854	1	195376	5	0,107500
6	18464	1	213840	6	0,117659
7	16508	1	230348	7	0,126742
8	16391	1	246739	8	0,135761
9	15827	1	262666	9	0,144470
10	14969	1	277535	10	0,152706
20	12311	1	413659	20	0,227604
30	7762	1	512051	30	0,281742
40	6866	1	584223	40	0,321452
60	5326	1	708632	61	0,389905
70	4183	1	756477	71	0,416230
80	3564	1	794862	81	0,437350
90	3217	1	828605	91	0,455917
100	2699	1	858548	101	0,472392
200	1214	1	1052304	209	0,579001
300	760	1	1160452	325	0,368506
400	518	1	1240985	456	0,682817
401	517	2	1242019	458	0,683386
500	366	2	1307201	609	0,719250
600	240	2	1377359	848	0,757853



Ранг частот R_f	Частота f	Кратность m	Накопленная частота A_f	Накопленная кратность A_m	Относительная накопл. част. C
700	137	27	1467679	1361	0,807549
794	43	31	1608826	3214	0,885211
806	31	47	1635882	3967	0,900098
827	10	500	1695851	7643	0,933094
828	9	426	1699685	8069	0,935204
829	8	692	1705221	8761	0,938250
830	7	681	1709988	9442	0,940873
831	6	1354	1718112	10796	0,945343
832	5	1971	1727967	12767	0,950765
833	4	3306	1741191	16073	0,958041
834	3	5053	1756350	21126	0,966382
835	2	13653	1783656	34779	0,981406
836	1	33793	1817449	68572	1,000000

Анализ содержания этой таблицы показывает, что число рангов частот наименований понятий коллекции документов содержит 836 рангов, а общее число наименований понятий в этой коллекции документов включает 1817449 понятий, из которых число разных понятий – всего 68572. При этом наименования понятий в коллекции распределены неравномерно. Так, например максимальная частота слова «проверка» имеет частоту 55255, в то же время более 32 тыс. слов и словосочетаний имеют частоту 1. В свете предлагаемых решений по созданию актуального семантического словаря было решено взять за основу словаря АСК данный частотный словарь, исключив из его состава высокочастотную и малочастотную части словаря. В процессе исследований было эмпирически установлено, что пороговым критерием для исключения высокочастотной лексики является значение частоты равное

$$\underline{f} = \frac{|D|}{3} = \frac{2013}{3} = 671, \quad (10)$$

где \underline{f} – высокочастотное пороговое значение; $|D|$ – число документов в коллекции.

Этому значению частоты в наибольшей степени соответствует ранг частоты R_f со значением 300 (325 понятий, с покрытием понятий коллекции документов на 63,8 %).

При формировании низкочастотного порогового значения ранга частоты, определяющего число исключенных из состава частотного словаря малочастотных понятий, необходимо исходить из значения минимального объема кластера. Если принять, что минимальным объемом кластера может быть кластер с объемом не менее 10 документов, тогда нижнее пороговое значения частот принимает равной 10:

$$\bar{f} = |\text{minclast}| = 10, \quad (11)$$

где \bar{f} – нижнее пороговое значение; $|\text{minclast}|$ – минимальное число документов в кластере. Этому критерию в данном случае будет соответствовать количество

разных понятий равное 7643 с покрытием понятий коллекции документов рангом частот более 10 на 93,3 %.

В этом случае объем АСК будет равен:

$$N = A_m [R_f(\bar{f})] - A_m [R_f(\underline{f})] = 7643 - 325 = 7318. \quad (12)$$

Покрытие понятий коллекции документов понятиями АСК будет равен 30 % (93,3 % - 63,8 = 29,5 %). Такое значение покрытия АСК соответствует эмпирически установленному оптимальному значению покрытия автоматически создаваемых словарей АСК для различных коллекций текстов. Всем элементам словаря назначались весовые коэффициенты по формулам (8, 9).

Проведенные эксперименты показали недостаточную смысловоразличающую способность АСК, обусловленную тем фактом, что семантически значимые для заданной тематике однословные понятия и аббревиатуры имеют неприемлемо низкий весовой коэффициент. Поэтому было решено увеличить вес смысловой значимости тех однословным понятиям, которые входят в состав разработанного в процессе данного исследования тематического концептуального словаря (ТКС) и переназначить в словаре АСК их весовые коэффициенты. Те понятия, которые были идентифицированы с элементами словаря, включались в состав АСК и им присваивались новые весовые коэффициенты смысловой значимости в соответствии с формулой:

$$W_{POS} = f * \begin{cases} W_{POS} = N^2, \text{ если } N > 1 \\ W_{POS} = 10, \text{ если } N = \frac{1}{K_{Abr}}, \\ W_{POS} = 10, \text{ если } N = K_{Sns} \end{cases}, \quad (13)$$

где f – частота; N – число слов в словосочетании; K_{Abr} – в словаре ТКС определено как аббревиатура; K_{Sns} – в словаре ТКС определено как значимое понятие предметной области.

Эксперименты, проведенные с модифицированным АСК с повышенными весовыми коэффициентами значимых понятий, показали его эффективность, увеличив вес ПОДов документов примерно на 15 %, и при этом существенно повысилась роль тех понятий, которые в наибольшей степени определяют основное смысловое содержание документов.

На втором этапе по текстам каждого документа коллекции был автоматически сформирован понятийный образ каждого документа (ПОД) коллекции. Формирование ПОДа осуществлялось путем выявления и идентификации в документе понятий, являющихся элементами АСК. При этом процесс идентификация понятий производилась на уровне словоизменительной нормализации слов, входящих в состав понятий.

Для каждого ПОДа документа вычислялась его семантическая характеристика (Под семантической характеристикой ПОДа понимается совокупность трех параметров: 1) число элементов ПОда; 2) число высокочастотных понятий; 3) вес ПОДа., которая использовалась при формировании первоначального списка кластерообразую-



щих документов коллекции. В список кластерообразующих документов включались документы, семантическая характеристика которых превышала пороговое значение.

На третьем этапе выполнялся процесс автоматического формирования кластеров коллекции документов. Формирование каждого кластера производилось путем сопоставления элементов ПОДов кластерообразующего документа с ПОДАми всей коллекции документов. В процессе такого сопоставления ПОДов вычислялось отношение суммы весов совпавших элементов ПОДа с весом ПОДа кластерообразующего документа. В зависимости от заданного порогового значения этого отношения принималось решение о включение документа коллекции в кластер данного кластерообразующего документа.

Для существенного уменьшения числа сравнений ПОДов документов коллекции в процессе кластеризации было принято следующее допущение:

Если процент совпадения весов ПОДов кластерообразующего документа и сравниваемого документа выше порогового (например, выше 30 %), то эти документы являются семантически близкими и в дальнейшем этот сравниваемый документ не должен использоваться в качестве кластерообразующего.

В рамках этих допущений было решено организовать два списка документов, изменяющихся в процессе кластеризации. Первый список («черный») должен включать документы, которые будут использовать в качестве кластерообразующих. Первоначально он сформирован на основе использования семантических характеристик ПОДа. В процессе кластеризации (формирования кластеров) из него должны исключаться те документы, которые уже использовались в качестве кластерообразующих и те документы, которые попали хотя бы в один из сформированных кластеров. Признаком завершения процесса кластеризации будет отсутствие документов в «черном» списке.

Одновременно для контроля процесса кластеризации будет создан постоянно пополняемый список («белый» список) документов, входящих хотя бы в один из сформированных кластеров. По завершению процесса кластеризации количество документов «белого» списка должно быть равно количеству документов коллекции.

Завершающим этапом процесса кластеризации являются операции нахождения центроида для каждого кластера, формирования названия кластера, а также проверка истинности процесса кластеризации. В процессе такой проверки должно быть установлено, что в каждом документе коллекции содержится обобщенное содержание кластера. Эту проверку можно осуществить путем формирования дайджеста кратких рефератов документов кластера. Все эти операции базируются на использовании автоматически создаваемого концептуального словаря кластера (КСК). Этот словарь строится для каждого кластера на основе анализа содержания документов, входящих в данный кластер. Поиск центроида осуществляется путем «взвешивания» вновь построенных ПОДов документов кластера по словарю КСК и удовлетворяющих двум условиям: 1) ПОД документ должен иметь максимальный вес, 2) источник документа должен быть включен в список верифицированных источников.



ТЕХНОЛОГИЧЕСКИЙ ПРОЦЕСС КЛАСТЕРИЗАЦИИ КОЛЛЕКЦИИ ДОКУМЕНТОВ

На основе предложенных решений был разработан обобщенный технологический процесс кластеризации, состоящий из следующих операций:

Этап 1. Конвертирование и формально-логический контроль каждого документа коллекции

- 1.1 Конвертирование исходных текстов во внутрисистемный формат.
- 1.2 Выполнение формально-логического контроля (ФЛК) текста.

Этап 2. Семантический анализ текста каждого документа коллекции

- 2.1 Графематический анализ текста.
- 2.2 Морфологический анализ слов текста.
- 2.3 Выделение наименований понятий текста.
- 2.4 Нормализация наименований понятий, выделенных из предложения.
- 2.5 Исключение незначимых слов и словосочетаний по словарю стоп-слов.
- 2.6 Построение списка формализованных наименований понятий документа с указанием частот их встречаемости.

Этап 3. Построение актуального концептуального словаря (АКС) для кластеризации коллекции документов

- 3.1 Создание массива всех формализованных наименований понятий с указанием частот их встречаемости в документах коллекции.
- 3.2 Построение частотного словаря формализованных наименований понятий коллекции документов.
- 3.3 Построение характеристической таблицы частотного словаря коллекции документов.
- 3.4 Определение верхнего и нижнего порогового значения частот словаря.
- 3.5 Формирование понятийного состава словаря АКС.
- 3.6 Назначение весового значения смысловой значимости наименованиям понятий словаря.
- 3.7 Корректировка значений весов понятий АКС по тематическому концептуальному словарю (ТКС).
- 3.8 Генерация машинного представления АКС.

Этап 4. Построение понятийного образа документа (ПОДа) коллекции

- 4.1 Выявление в тексте наименований понятий по словарю АКС.
- 4.2 Формирования ПОДа документа.
- 4.3 Назначение каждому элементу ПОДа весовых характеристик.

Этап 5. Кластеризация коллекции текстов

- 5.1 Получение информации о семантических характеристиках ПОДов документов коллекции.
- 5.2 Формирование первоначального «черного» списка документов коллекции.
- 5.3 Последовательное сопоставление ПОДов «черного» списка документов с документами коллекции.



5.4 Ведение «черного» и «белого» списков документов коллекции в процессе кластеризации.

5.5 Формирование кластеров на основе анализа результатов сопоставление ПОДов документов.

Этап 6. Установление центроида кластера

6.1 Выделение наименований понятий из документов кластера.

6.2 Нормализация наименований понятий, выделенных из предложения.

6.3 Исключение незначимых слов и словосочетаний по словарю стоп-слов.

6.4 Построение списка формализованных наименований понятий документа с указанием частот их встречаемости.

6.5 Формирование понятийного состава концептуального словаря кластера (КСК).

6.6 Назначение значений коэффициентов смысловой значимости наименованиям понятий словаря КСК.

6.7 Корректировка значений весов понятий словаря КСК по ТКС.

6.8 Генерация машинного представления КСК.

6.9 Формирования ПОДа документов кластера по словарю КСК.

6.10 Определение веса ПОДов документа кластера.

6.11 Определение центроида кластера.

Этап 7. Генерация обобщенного содержания кластера в виде дайджеста рефератов

7.1 Определение местоположения наименований понятий в предложениях документов кластера.

7.2 Вычисление суммарных весов предложений документа.

7.3 Установление заданных N предложений с максимальным весом.

7.4 Построение краткого реферата каждого документа, состоящего из заголовка и N заданных предложений с максимальным весом.

7.5 Формирование дайджеста обобщенных рефератов документов кластера.

На основе предложенных теоретических решений было разработано экспериментальное программное обеспечение. Разработанная программно-технологическая схема процесса кластеризации документов приведена на рис. 1. Представленная схема состоит из программного модуля хранения декларативных средств и семи последовательно выполняемых программных модулей:

1. Модуль конвертирования и ФЛК коллекции текстов.

2. Модуль семантического анализ текста документа.

3. Модуль построения АКС.

4. Модуль построения ПОДа.

5. Модуль кластеризации коллекции текстов.

6. Модуль установления центроида кластера.

7. Модуль генерации обобщенного содержания кластера.

В свою очередь, как видно на рис. 1, каждый программный модуль состоит из нескольких функциональных модулей, выполняющих конкретную целевую функцию. Реализация файлового обмена между программными модулями и модулями позволяет в полной мере контролировать процесс кластеризации на различных его этапах.

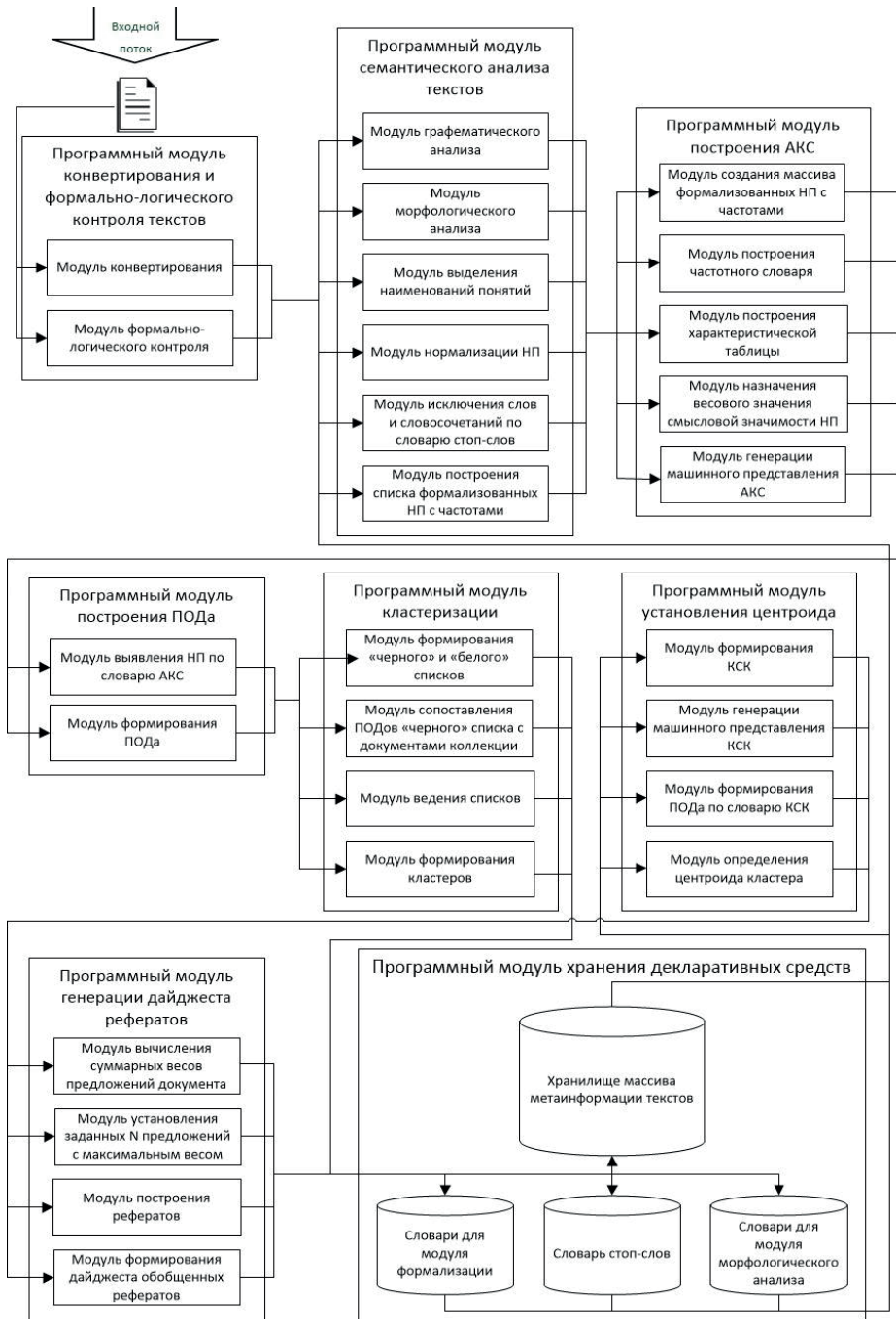


Рис. 1. Система кластеризации документов



ЗАКЛЮЧЕНИЕ

В статье описано решение задачи автоматической кластеризации документов СМИ на основе их смыслового анализа. Предлагаемое решение базируется на методах машинной грамматики, семантико-синтаксического и концептуального анализа текстов, а также на методах выявления понятийного состава коллекции документов и формализации смыслового содержания текстов. Автоматическое выявление в текстах наименований понятий (сущностей) базируется на методах концептуального анализа текстов и методике вычисления меры их смысловой значимости, использующей статистические, синтаксические и семантические признаки. На основе предложенных методов разработаны алгоритмы и программное обеспечение автоматической кластеризации документов СМИ на основе их смыслового анализа. Проведенные эксперименты показали работоспособность предложенных решений автоматической кластеризации коллекции документов и возможность их использования в промышленных реализациях.

Литература

1. *Богатырев М.Ю.* Извлечение фактов из текстов естественного языка с применением концептуальных графовых моделей // Известия ТулГУ. Технические науки. – 2016. – № 7. – Ч. 1.
2. Современные технологии обработки естественного языка в задачах стратегического управления / Виноградов А.Н. [и др.] // Технологическая перспектива в рамках евразийского пространства: новые рынки и точки экономического роста. / Власова Н.А., Куршев Е.П., Подобрыв А.В. – СПб.: Центр научно-информационных технологий «Астерион», – 2018.
3. *Ермаков А.Е.* Автоматическое извлечение фактов из текстов досье: опыт установления анафорических связей [Электронный ресурс] // Компьютерная лингвистика и интеллектуальные технологии: труды Международной конференции «Диалог'2007». – М. : Наука, – 2007.
4. Автоматическое создание формализованного представления смыслового содержания неструктурированных текстовых сообщений СМИ и социальных сетей / Хорошилов Ал-др. А. [и др.] // Системы высокой доступности / Никитин Ю.В., Хорошилов Ал-ей. А., Будско В.И., – 2014. – Т. 10., № 3.
5. *Helbig H.* Knowledge representation and the semantics of natural language. – Berlin: Springer, – 2006.
6. *Белоногов Г.Г., Гиляревский Р.С., Хорошилов А.А.* Проблемы автоматической смысловой обработки текстовой информации // Научно-техническая информация. Сер. 2. Информационные процессы и системы / Всероссийский институт научной и технической информации РАН. – 2012, № 11. – С. 24–28.
7. Средства машинной грамматики русского языка (по Г.Г. Белоногову) / Аблов И.В. [и др.] // Научно-техническая информация / Козичев В.Н., Ширманов А.В., Хорошилов Ал-др А., Хорошилов Ал-ей А., Сер. 2, – 2018. № 6.
8. *Калинин Ю.П., Хорошилов Ал-др. А., Хорошилов Ал-ей. А.* Современные технологии автоматизированной обработки текстовой информации // Системы высокой доступности, – 2015. – Т. 11, № 2.



Automatic Clustering of Mass Media Documents Based on the Analysis of Their Semantic Content

Anna V. Kan*

Moscow Aviation Institute (National Research University), Moscow, Russia
ORCID: <https://orcid.org/0000-0001-9410-406X>
e-mail: avkan@nrczh.ru

Yana D. Kozlovskaya**

Moscow Aviation Institute (National Research University), Moscow, Russia
ORCID: <https://orcid.org/0000-0002-1780-5687>
e-mail: yana_kozlovskaja@mail.ru

Nikolay A. Kadushkin***

Moscow Aviation Institute (National Research University), Moscow, Russia
ORCID: <https://orcid.org/0000-0002-0327-909X>
e-mail: bbamrin@gmail.com

Aleksandr A. Khoroshilov****

State Scientific Research Institute of aircraft systems (GosNIIAS), Moscow, Russia
ORCID: <https://orcid.org/0000-0003-4885-3232>
e-mail: aleksandr_khor@mail.ru

The article describes the solution to the problem of automatic clustering of media documents based on the analysis of their semantic analysis. The proposed solution is based on the methods of machine grammar, semantic-syntactic and conceptual analysis of texts, as well as methods for identifying the conceptual composition of a collection of documents and formalizing the semantic content of texts. The developed algorithm of the document clustering process provides for the possibility of its implementation in a fully automatic mode without prior machine learning.

Keywords: automatic clustering of documents, machine grammar, semantic-syntactic analysis of texts, conceptual analysis of texts, actual conceptual vocabulary.

For citation:

Kan A.V., Kozlovskaya Y.D., Kadushkin N.A., Khoroshilov Al-dr.A. Automatic Clustering of Mass Media Documents Based on the Analysis of Their Semantic Content. *Modelirovanie i analiz dannykh = Modelling and Data Analysis*, 2020. Vol. 10, no. 3, pp. 24–38. DOI: <https://doi.org/10.17759/mda.2020100302> (In Russ., abstr. in Engl.).

***Anna V. Kan**, associate professor of the Institute of Moscow Aviation Institute (National Research University), Moscow, Russia, ORCID: <https://orcid.org/0000-0001-9410-406X>, e-mail: avkan@nrczh.ru

****Yana D. Kozlovskaya**, student of the Institute of Moscow Aviation Institute (National Research University), Moscow, Russia, ORCID: <https://orcid.org/0000-0002-1780-5687>, e-mail: yana_kozlovskaja@mail.ru

*****Nikolay A. Kadushkin**, student of the Institute of Moscow Aviation Institute (National Research University), Moscow, Russia, ORCID: <https://orcid.org/0000-0002-0327-909X>, e-mail: bbamrin@gmail.com

******Aleksandr A. Khoroshilov**, engineer of State Scientific Research Institute of aircraft systems (GosNIIAS), Moscow, Russia, ORCID: <https://orcid.org/0000-0003-4885-3232>, e-mail: aleksandr_khor@mail.ru



References

1. Bogatyrev M. Yu. Izvlechenie faktov iz tekstov estestvennogo yazyka s primeneniem kontseptual'nykh grafovyykh modelei [Fact extraction from natural language texts with conceptual graph models]. *Izvestiya TulGU. Tekhnicheskie nauki*. – 2016. – № 7. – Ch. 1.
2. Vinogradov A.N. [i dr.] *Sovremennye tekhnologii obrabotki estestvennogo yazyka v zadachakh strategicheskogo upravleniya* [Modern technologies of natural language processing in strategic management tasks]. *Tekhnologicheskaya perspektiva v ramkakh evraziiskogo prostranstva: novye rynki i tochki ekonomicheskogo rosta*. – SPb.: Tsentr nauchno-informatsionnykh tekhnologii “Asterion”, 2018.
3. Ermakov A.E. *Avtomaticheskoe izvlechenie faktov iz tekstov dos'je: opyt ustanovleniya anaforicheskikh svyazei* [Elektronnyi resurs] [Automatic extraction of facts from dossier texts: an experience of establishing anaphoric connections]. *Komp'yuternaya lingvistika i intellektual'nye tekhnologii: trudy Mezhdunarodnoi konferentsii «Dialog'2007»*. – Moscow. : Nauka, 2007.
4. Khoroshilov Al-dr. A. [i dr.] *Avtomaticheskoe sozdanie formalizovannogo predstavleniya smyslovogo sodержaniya nestrukturirovannykh tekstovykh soobshchenii SMI i sotsial'nykh setei* [Automatic creation of a formalized representation of the semantic content of unstructured text messages of the media and social networks]. *Sistemy vysokoi dostupnosti*, № 3, Vol. 10, 2014.
5. Helbig H. *Knowledge representation and the semantics of natural language*. – Berlin: Springer, 2006.
6. Belonogov G.G., Gilyarevskii R.S., Khoroshilov A.A. *Problemy avtomaticheskoi smyslovoi obrabotki tekstovoi informatsii* [Problems of automatic semantic processing of text information]. *Nauchno-tekhnicheskaya informatsiya. Ser. 2. Informatsionnye protsessy i sistemy / Vserossiiskii institut nauchnoi i tekhnicheskoi informatsii RAN*. – 2012, № 11. – pp. 24–28.
7. Ablov I.V. [i dr.] *Sredstva mashinnoi grammatiki russkogo yazyka (po G.G. Belonogovu)* [Means of machine grammar of the Russian language (according to G.G. Belonogov)]. *Nauchno-tekhnicheskaya informatsiya. Ser. 2, № 6*, 2018.
8. Kalinin Yu.P., Khoroshilov Al-dr. A., Khoroshilov Al-ei. A. *Sovremennye tekhnologii avtomatizirovannoi obrabotki tekstovoi informatsii* [Modern technologies for automated processing of text information]. *Sistemy vysokoi dostupnosti*, № 2, Vol. 11, 2015.



УДК 316.6

Модель процесса адаптивного обучения и его программная реализация

Поминов Д.А.*

Московский государственный психолого-педагогический университет
(ФГБОУ ВО МГППУ), г. Москва, Российская Федерация
ORCID: <https://orcid.org/0000-0002-1321-3713>
e-mail: pominovda@mgppu.ru

Представленный проект направлен на автоматизацию процесса электронного обучения, касающуюся приобретения практических навыков для решения неформализуемых задач, определения уровня знаний и сокращения продолжительности тестирования за счет уменьшения количества задач в зависимости от уровня подготовки. Для удовлетворения указанных выше требований был реализован подход адаптивного тестирования и реализован веб-сервис «Адаптивный тренажер» для демонстрации его работы.

Ключевые слова: адаптивное обучение, марковские случайные процессы, адаптивный тренажер, самообучающиеся системы, разработка информационных систем.

Для цитаты:

Поминов Д.А. Модель процесса адаптивного обучения и его программная реализация // Моделирование и анализ данных. 2020. Том 10. № 3. С. 39–52. DOI: <https://doi.org/10.17759/mda.2020100303>

Финансирование. Работа выполнена при поддержке ФГБОУ ВО МГППУ в рамках проекта «Исследования по адаптивному обучению: разработка и внедрение веб-ресурса “Самообучающийся адаптивный тренажер для подготовки абитуриентов к экзаменам по математике” и новой концепции адаптивного обучения».

**Поминов Денис Александрович*, младший научный сотрудник, факультет информационных технологий, Московский государственный психолого-педагогический университет (ФГБОУ ВО МГППУ), г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0002-1321-3713>, e-mail: pominovda@mgppu.ru



ВВЕДЕНИЕ

В силу особенностей организации процесса обучения в школьных классах, зачастую ученикам не хватает времени урока на решение задач под контролем учителя. В результате прибегают к услугам репетиторов или к самообучению. Для осуществления самостоятельной подготовки в последние годы становятся популярными различные образовательные онлайн сервисы. У этих систем есть некоторые недостатки, одни из которых – это невозможность индивидуальной настройки под конкретного пользователя, неполная автоматизация, жесткая негибкая структура процесса тестирования. Существующие системы в области электронного обучения [13, 15, 17, 20] в основном решают более простые задачи. Это связано в первую очередь с отсутствием подходящего математического аппарата и сложностями в реализации подобных систем.

В данной работе предложен подход для решения подобных задач, основанный на концепте адаптивного обучения с использованием марковских процессов [2–3, 5–7, 10, 12, 18–19].

Этот подход является альтернативным адаптивным технологиям основанных на современной IRT (Item Response Theory) [14, 16, 22–24]. Базовый тезис сформулирован Г. Рашем [21]: вероятность правильного ответа на задание определяется различием в оценках сложности теста, уровня знаний, умений или других конструктов и выражается функцией сигмоидного типа. Практическое использование этой идеи имеет некоторые ограничения: статичность оценок, большое количество заданий, трудности при построении оценок и динамики времени, затраченного на решение задач.

Одна из главных проблем адаптивного выбора задач (основанных на оценках IRT) связана с примерным равенством вероятности правильного и неправильного выполнения задач. Это делает результаты тестирования в основном зависимыми от внешних и случайных факторов, которые не связаны с измеряемыми конструктами.

В 2010–2012 был разработан метод адаптивного тестирования [8–9, 11], основанный на использовании идентифицируемых марковских моделей с непрерывным временем и байесовским классификатором. В развитие этого результата в 2017 г. была предложена новая версия марковской модели адаптивного тестирования с дискретным временем [1], которая предполагает оценки конструктов с использованием предельных распределений вероятностей нахождения в состояниях, рассчитанных с использованием матриц вероятностей перехода. Этот подход можно рассматривать как расширение IRT, поскольку в качестве компонента используется модель Г. Раша.

Созданные марковские модели адаптивного тестирования стали основой для разработки адаптивного тренажера [4] для обучения неформализованным навыкам и умениям, необходимым для решения математических и других задач достаточно высокой сложности, требующих овладения как стандартной методикой построения рассуждений, так и элементами творческого мышления.

В первую очередь адаптивный тренажер считается наиболее эффективным в случаях, когда необходимо упорядочить полученные знания и навыки, например, для подготовки к экзаменам при решении неформализуемых задач (математические,

технические, алгоритмические и т.п.). Также важно заметить, что данный сервис не является заменой учителя, а скорее дополняет существующий процесс обучения, расширяя возможности самоподготовки.

ОПИСАНИЕ МАТЕМАТИЧЕСКОЙ МОДЕЛИ

Подбор заданий во время тестирования осуществляется с помощью параметрических математических моделей, которые описываются марковскими случайными процессами с дискретными состояниями и непрерывным временем. Динамика изменения вероятностей пребывания в состояниях этих процессов определяется системой уравнений Колмогорова [4]:

$$\frac{dp(t)}{dt} = M(\lambda)p(t),$$

где $0 \leq t \leq T$, $p(t)$ – вероятности пребывания в состояниях процесса, λ – множество интенсивностей переходов между состояниями, M – матрица интенсивностей переходов между состояниями. Значения указанных интенсивностей определяются начальными распределениями вероятностей и наблюдаемыми частотами пребывания в состояниях $F_{i,d} \{F_i\}_{i=0,\dots,n-1}$ в моменты времени $\{t_d\}_{d=0,\dots,D-1}$, где i – индексы состояний марковского процесса; D – количество моментов времени, в которые фиксировались частоты $F_{i,d}$; $0 \leq t_d \leq T$; T – конечный момент времени [5].

Для описания того, как вероятности нахождения в заданных состояниях изменяются со временем, применяются процессы, организованные по схеме, которая представлена на Рис. 1.

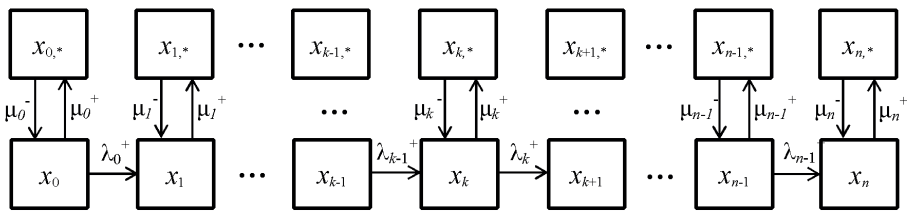


Рис. 1. Сеть Маркова, описывающая процесс тестирования с непрерывным временем: $\{x_i\}_{i=0,\dots,n}$ и $\{x_i^*\}_{i=0,\dots,n}$ – состояния марковского процесса, $\lambda = (\lambda_0^+, \dots, \lambda_{n-1}^+, \mu_0^+, \dots, \mu_n^+, \mu_0^-, \dots, \mu_n^-)^T$ – интенсивности переходов между состояниями

Перемещения между состояниями сети Маркова (Рис. 1) определяются следующими правилами:

1. Если ученик находится в состоянии x_k , успешно выполнил задание и уложился по времени, то он переходит в состояние x_k^* ;
2. Если ученик находится в состоянии x_k , не выполнил задание, но уложился по времени, то он остается в состоянии x_k ;



3. Если ученик находится в состоянии x_k , успешно выполнил задание, но не уложился по времени, то он переходит в состояние x_{k*} ;
4. Если ученик находится в состоянии x_{k*} , правильно решает задание и не укладывается по времени или неправильно решает задание и укладывается по времени, то он остается в состоянии x_{k*} ;
5. Если ученик находится в состоянии x_{k*} , успешно выполнил задание и уложился по времени, то он переходит в состояние x_k .

Тренировка начинается с получения задания (выбирается случайно), которое соответствует минимальному уровню (состояние x_0). Далее задания предъявляются в зависимости от того, как справляется с ними ученик. Например, если он быстро дает неправильные ответы, то сложность не повышается, и он остается на прежнем уровне; если он укладывается по времени и в основном дает правильные ответы, то сложность постепенно увеличивается. По завершению ученик оказывается в состоянии, которое соответствует его уровню навыка. Тренаж завершается по достижению лимита (общего времени на всю тренировку) или при успешном выполнении задания за отведенное время в конечном состоянии сети Маркова x_n .

Оценка вероятности нахождения в определенном состоянии (принадлежности к некоторому уровню подготовки) производится с помощью формулы Байеса:

$$P(S) = \frac{P(C_i)P(S|C_i)}{\sum_{k=0}^z P(C_k)P(S|C_k)},$$

где C_i – событие, связанное с наличием у тестируемого i -го уровня подготовки ($i=0, \dots, z$), S – событие, связанное с нахождением в заданном состоянии модели в указанный момент времени при указанном уровне трудности заданий, $P(C_i)$ – априорная вероятность появления i -го уровня подготовки у тестируемого, $P(S|C_i)$ – вероятность нахождения в заданном состоянии эталонной модели в указанный момент времени при наличии i -го уровня подготовки, $P(S)$ – вероятность i -го уровня подготовки при условии нахождения в заданном состоянии модели в указанный момент времени [5].

Более подробно математическая модель и принцип работы адаптивного тренажера и адаптивного тестирования представлены в работах [1–7, 10, 12, 18–19].

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ АДАПТИВНОГО ТРЕНАЖЕРА

Программная реализация адаптивного тренажера условно делится на две части: интерфейсная (пользовательская) и серверная. Коммуникация и синхронизация информации между ними реализована с использованием технологии AJAX, JSON и API модуля, обрабатывающего входящие запросы. Серверная часть написана на языке программирования PHP и отвечает за обработку данных: получение задания, проверка правильности ответа, подсчет параметров математической модели (вероятностей) и т.п. Интерфейсная часть реализована с помощью веб фреймворка Laravel и reactJS. Также кроме интерфейса она имеет свою базу данных, где хранятся данные по сессии.



ям (время начала и конца тестирования, время на ответ, правильность ответа) и пользователям (логин, почта, пароль, роль). Таким образом, ядро адаптивного тренажера, включающее всю обработку и непосредственно саму базу данных заданий, отделено от реализации интерфейса, с которым взаимодействует пользователь.

Режим работы с тренажером следующий: пользователь регистрируется в системе, ему присваивается по умолчанию роль ученика, он авторизуется, после чего может выбрать раздел для тренировки. Перед началом тренажа пользователь может ознакомиться со справочной информацией, встроенной в тренажер, и узнать каким образом решать задания по выбранной теме. Каждый раздел делится на несколько содержательных уровней (по 4 в теме) и 3 сложности.

Во время прохождения тренажа интерфейс разделен на несколько блоков (Рис. 2): текст задания, варианты ответа, история прохождения (номер вопроса, трудность от 1 до 3, верный или неверный ответ), таймер, номер задания с указанием сложности. Пользователь непосредственно взаимодействует только с блоком ввода ответа (вид которого зависит от типа задания). По завершению решения задачи на сервер с помощью AJAX запроса отправляются данный учеником ответ вместе с идентификатором вопроса и временем, затраченным на решение. На стороне сервера происходит первичная обработка ответа (удаляются пробелы и спецсимволы и т.п.), и в зависимости от типа задания (точная сверка, дроби и вычисления, тригонометрия, интервалы, выбор из вариантов и пр.) проверяется его правильность с полученным из БД “Trainer” контрольным вариантом. Далее анализируется история прохождения тестирования в соответствии с правилами и моделью, которые рассматривались выше. В результате в качестве ответа сервер отправляет клиенту либо номер следующего задания, либо отметку о завершении тестирования.

0:09

Тренажер по математике

№1 Какой знак имеет произведение корней уравнения:
 $x^2 + 3x - 4 = 0$?

Уровень 1

отрицательный

положительный

Рис. 2. Интерфейс адаптивного тренажера



По окончании тестирования пользователю будет отображено (Рис. 3): процент правильных ответов, время тренировки, сравнение с прошлой тренировкой (если такая была). Также он может ознакомиться с результатами всех остальных своих тренировок.

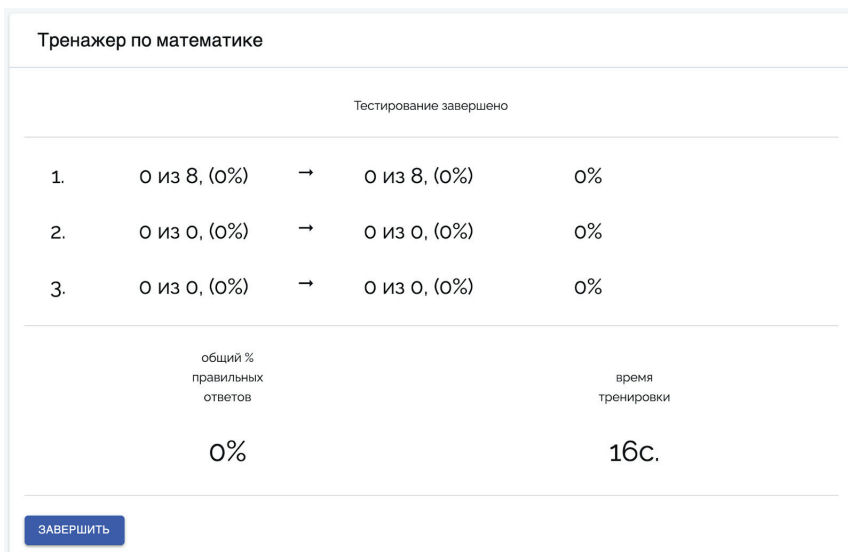


Рис. 3. Интерфейс адаптивного тренажера

В информационной системе тренажера определено 3 роли: ученик (student), преподаватель (teacher), администратор (admin).

Преподаватель может объединять учеников в группы, просматривать задания и статистику прохождения тестирования (Рис. 4, 5). Администратор выполняет вспомогательную роль (Рис. 6): сброс паролей, редактирование личной информации (логин, почта, имя), изменение роли пользователя.

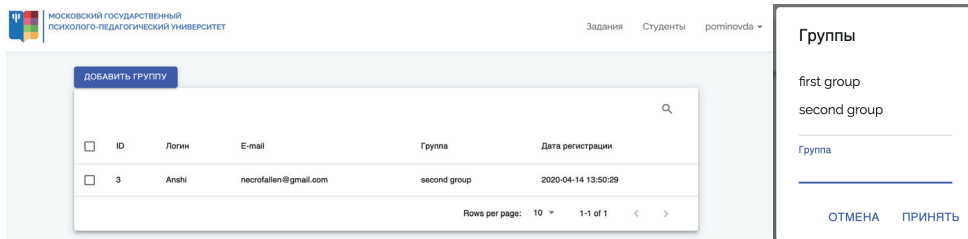


Рис. 4. Личный кабинет преподавателя



PREV NEXT CHECK Раздел Тема Сложно

Квадратные уравнения и н... Квадратное уравнение. Формула корней квадратного урав... 2

ID : 15 Статус : Проверено
pos : 1 Section : Квадратные уравнения и неравенства
2,3,2 Theme : Квадратное уравнение. Формула корней квадратного уравнения (Уровень - 3)

Решить уравнение $x^2 + 4x - 5 = 0$.

-5, 1

1, 5
 -1, -5
 -5, 1
 3, 2

Рис. 5. Личный кабинет преподавателя

ID	Логин	E-mail	Роль	Дата регистрации		
<input type="checkbox"/>	1	admin	necrofallen@hotmail.com	admin	2019-04-24 09:14:05	Пользователь ID 2 pominovda Дата регистрации 2019-05-06 07:47:26
<input type="checkbox"/>	2	pominovda	pominovda@mgppu.ru	teacher	2019-05-06 07:47:26	Login name pominovda Email Address pominovda@mgppu.ru
<input type="checkbox"/>	3	Anshi	necrofallen@gmail.com	student	2020-04-14 13:50:29	Роль Преподаватель ▾

Rows per page: 10 ▾ 1-3 of 3 < >

ОТМЕНА ПРИНЯТЬ

Рис. 6. Личный кабинет администратора

Информационная система адаптивного тренажера предполагает отслеживание статистических данных по тренировкам (Рис. 7, Табл. № 1), например:

1. общее кол-во тренировок по каждому пользователю,
2. кол-во тренировок пользователя по теме,
3. распределение по пройденным темам и содержательным уровням.

Статистика тренировок					
#	Раздел	Кол-во вопросов	% правильных ответов	Время (мин)	Дата
1	Текстовые задачи	12	7 (58,33%)	1,92	2019-06-13 14:23:23
2	Квадратные уравнения и неравенства	2	0 (0%)	0,18	2019-06-18 00:00:00
3	Квадратные уравнения и неравенства	1	0 (0%)	0	2019-06-18 00:00:00
4	Квадратные уравнения и неравенства	16	2 (12,5%)	2371,25	2019-06-18 09:04:00
5	Текстовые задачи	6	0 (0%)	0,97	2019-06-18 09:07:14

Рис. 7. Статистика по тренировкам.



Таблица 1

Статистика по тренировкам

Сложность	1	2	3
Кол-во вопросов	2	9	0
% правильных ответов	50 %	11 %	0 %
Среднее время на ответ, с.	149	662	0
Пропуск ответа	1	6	0
Общее время тренажа	895 с (14 мин 55 сек)		

База данных делится на две части: БД Laravel Рис. 8 (вспомогательные системные таблицы, сессии, данные пользователей, данные тренировок, обратная связь, роли, группы), БД Trainer Рис. 9 (задания, разделы, содержательные уровни, коэффициенты, варианты проверки).

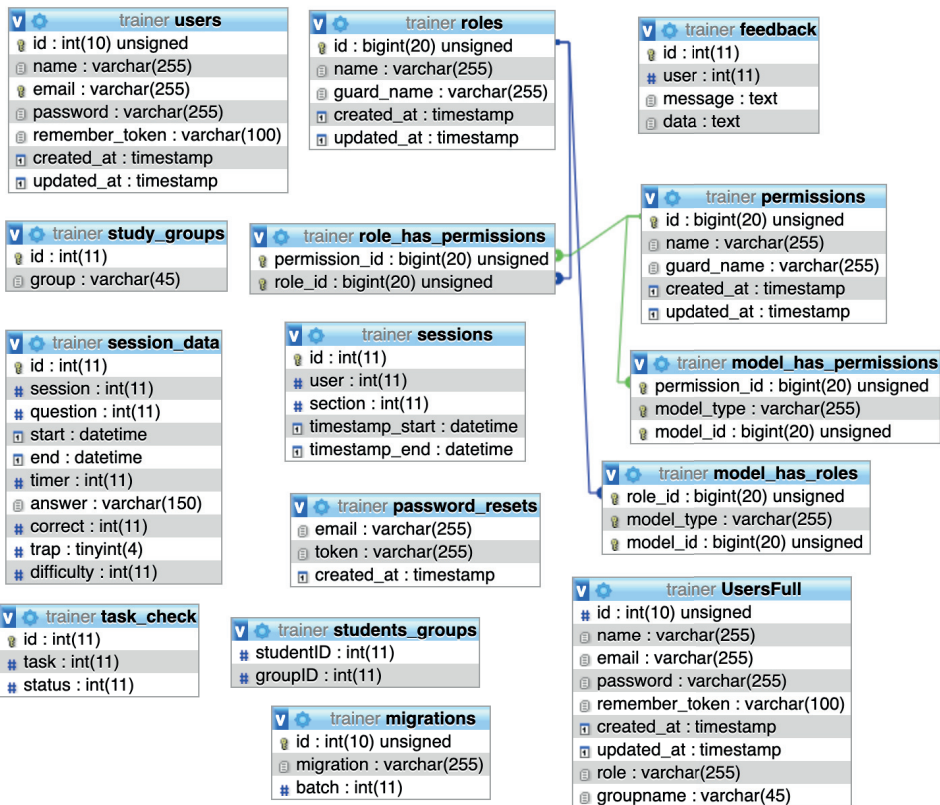


Рис. 8. Структура БД Laravel

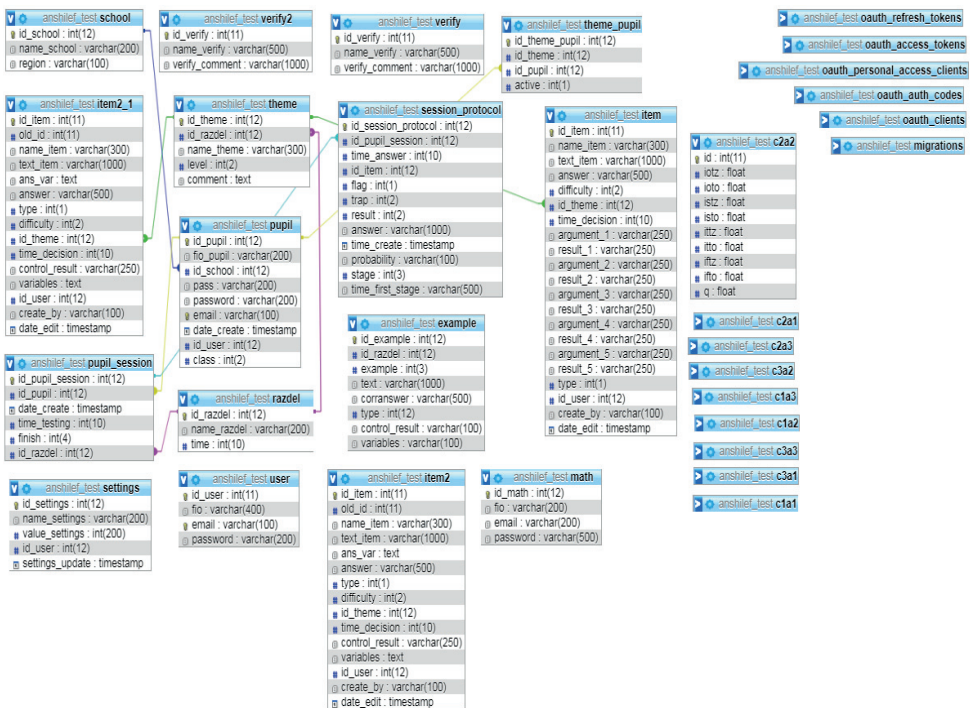


Рис. 9. Структура БД Trainer

Авторизация в системе адаптивного тренажера происходит с использованием БД “Laravel”, где хранятся данные пользователей и пароли в зашифрованном виде. Данные по доступным разделам загружаются с помощью Ajax запроса и ответа от сервера в формате JSON, который соответствует БД “Trainer”. Основными сущностями ее являются: razdel (список разделов), theme (содержательные уровни и справочная информация по каждому из них), item (данные по заданиям: текст задания, тип, ответ, сложность), c1a1 – c3a3 (матрицы коэффициентов), справочник типов заданий.

После выбора раздела и начала тестирования, система регистрирует новую сессию в БД Laravel. Каждой сессии (таблица sessions) соответствуют следующие данные: идентификатор, раздел, дата/время начала и завершения. Данные по прохождению тренировки записываются в таблицу session_data: идентификатор сессии, идентификатор задания, время начала и завершения, время ответа в секундах, текст ответа, корректность (верно или неверно), нахождение в состоянии ловушки, сложность задания. По окончании тестирования или по достижению временного лимита в таблице sessions ставится отметка о завершении.

Для управления правами доступа к различным интерфейсам системы используют модуль для Laravel – Spatie и соответствующие сущности в БД: roles, permissions, model_has_permissions и др. Он позволяет гибко настраивать роли и права для каждой модели поведения пользователя информационной системы.



ПРИМЕР РАБОТЫ АДАПТИВНОГО ТРЕНАЖЕРА

В качестве демонстрации работы адаптивного тренажера рассмотрим несколько сценариев использования, которые условно будут соответствовать разным уровням подготовки учеников: низкий, средний, высокий. Также рассмотрим состояние «ловушки», когда ученик не уложился в определенный интервал времени, отведенный на решение задачи. Сравнение тренировок представлено в Табл. 2.

Таблица 2

Сравнение тренировок

№	Кол-во заданий по трудностям, верно/неверно			Состояние «ловушки»
	1	2	3	
1	0 из 8	0	0	нет
2	2 из 2	3 из 3	4 из 4	нет
3	5 из 5	3 из 3	4 из 4	В 1 вопросе
4	2 из 2	3 из 3	4 из 8	нет

В первом случае давались «быстрые» (укладываются в заданный промежуток времени в зависимости от сложности вопроса) неправильные ответы. При таком сценарии ученик остается на первом уровне сложности, ему предъявляются первые 8 заданий из начального содержательного уровня, после чего тренировка завершается без повышения трудности. Во втором случае, ученик дал 5 правильных ответов подряд на первом и втором уровне, уложившись в заданный промежуток времени. В результате тренажер дважды повысил трудность заданий до третьего самого высокого уровня. В третьем, также давались только правильные ответы, но в первом вопросе ученик вышел за отведенный интервал на решение (5 минут) и попал в состояние «ловушки». В результате количество заданий на первом уровне повысилось. В последнем сценарии затруднения возникли только на третьей трудности. Итоговое количество задание указано в Табл. 3.

Таблица 3

Итоговое количество заданий по различным сценариям

№	Кол-во предъявленных заданий
1	8
2	9
3	12
4	13

Во всех случаях количество заданий напрямую зависит от уровня подготовки ученика. Также важно заметить, что для однозначного определения к некоторому классу (владения навыком) системе не требуется большое количество заданий.

ЗАКЛЮЧЕНИЕ

Создана информационная система «Адаптивный тренажер для подготовки абитуриентов к экзаменам по математике» и новый подход к тестированию на основе марковских процессов с дискретными состояниями и непрерывным временем.



Данный подход позволяет сократить время тренажа за счет уменьшения количества заданий и увеличить эффективность применения практических навыков в решении математических задач учениками школ при подготовке к экзаменам.

Литература

1. *Куравский Л.С., Артеменков С.Л., Юрьев Г.А., Григоренко Е.Л.* Новый подход к компьютеризированному адаптивному тестированию // Экспериментальная психология. 2017. Т. 10. № 3. С. 3345. doi:10.17759/exppsy.2017100303
2. Патент на полезную модель № 118095 (РФ). Устройство для моделирования адаптивного тестирования когнитивных способностей испытуемого / *Л.С. Куравский, С.Д. Кулик, П.А. Мармалюк, Г.А. Юрьев*. Заявка № 2012105993/08, 21.02.2012; Зарегистр. 10.07.2012; Опубликовано 10.07.2012 Бюл. № 19; Приоритет от 21.02.2012(РОСПАТЕНТ).
3. *Куравский Л.С., Марголис А.А., Мармалюк П.А., Юрьев Г.А., Думин П.Н.* Обучаемые марковские модели в задачах оптимизации порядка предъявления психологических тестов // Нейрокомпьютеры: разработка и применение. 2013. № 4. С. 28–38.
4. *Куравский Л.С., Марголис А.А., Мармалюк П.А., Панфилова А.С., Юрьев Г.А.* Математические аспекты концепции адаптивного тренажера // Психологическая наука и образование. 2016. Т. 21. № 2. С. 84–95. doi: 10.17759/pse.2016210210.
5. *Куравский Л.С., Марголис А.А., Юрьев Г.А., Поминов Д.А.* Концепция самообучающегося адаптивного тренажера // Нейрокомпьютеры: разработка и применение. 2018. № 3. С. 29–37.
6. Патент на полезную модель № 122796 (РФ). Система поддержки принятия решений для психологического и педагогического тестирования / *Л.С. Куравский, А.А. Марголис, П.А. Мармалюк, Г.А. Юрьев, П.Н. Думин, С.Д. Кулик*. Заявка № 2012132684/08, 31.07.2012; Опубликовано 10.12.2012, Бюл. № 34; Приоритет от 31.07.2012 (РОСПАТЕНТ).
7. *Куравский Л.С., Марголис А.А., Юрьев Г.А., Мармалюк П.А.* Концепция системы поддержки принятия решений для психологического тестирования // Психологическая наука и образование. 2012. № 1. С. 5665.
8. *Куравский Л.С., Юрьев Г.А.* Адаптивное тестирование как марковский процесс: модели и их идентификация // Нейрокомпьютеры: разработка и применение. 2011. № 2. С. 2129.
9. *Куравский Л.С., Юрьев Г.А.* Вероятностный метод фильтрации артефактов при адаптивном тестировании // Экспериментальная психология. 2012. Т. 5. № 1. С. 119131.
10. *Куравский Л.С., Юрьев Г.А.* Использование марковских моделей при обработке результатов тестирования // Вопросы психологии. 2011. № 2. С. 98107.
11. *Куравский Л.С., Юрьев Г.А.* Об одном подходе к адаптивному тестированию и устранению его артефактов // Нейрокомпьютеры: разработка и применение. 2012. № 1.
12. Марковские модели в задачах диагностики и прогнозирования: Учеб. пособие / Под ред. Л.С. Куравского. Изд. 2-е, доп. М.: Изд-во МГППУ. 2017. 203 с.
13. 1С: Электронное обучение [Электронный ресурс] // URL: <http://v8.1c.ru/elo> (дата обращения 06.02.2016).
14. *Baker F.B.* The Basics of Item Response Theory // ERIC Clearinghouse on Assessment and Evaluation. University of Maryland, College Park. MD. 2001.
15. Ebmodo [Электронный ресурс] // Connect with students and parents in your paperless classroom. URL: <https://www.edmodo.com> (дата обращения 03.02.2016).
16. *Gregory R.J.* Psychological testing: History, principles, and applications (5th edition). New York: Pearson. 2007.
17. *Kats Y.* Learning Management Systems and Instructional Design: Best Practices in Online Education. IGI Global. 2013. ISBN 9781466639317



18. Kuravsky L.S., Marmalyuk P.A., Yuryev G.A., Dumin P.N. A Numerical Technique for the Identification of Discrete-State Continuous-Time Markov Models // *Applied Mathematical Sciences*. 2015. V. 9. № 8. P. 379–391. URL: <http://dx.doi.org/10.12988/ams.2015.410882>.
19. Kuravsky L.S., Marmalyuk P.A., Baranov S.N., Alkhimov V.I., Yuryev G.A., Artyukhina S.V. A New Technique for Testing Professional Skills and Competencies and Examples of its Practical Applications // *Applied Mathematical Sciences*. 2015. V. 9. № 21. P. 1003–1026. <http://dx.doi.org/10.12988/ams.2015.411899>.
20. Moodle open-source learning platform [Электронный ресурс] // Moodle Pty Ltd. URL:<https://moodle.org> (дата обращения 03.02.2016).
21. Rasch G. Probabilistic models for some intelligence and attainment tests // Copenhagen, Danish Institute for Educational Research, expanded edition (1980) with foreword and afterword by B.D. Wright. Chicago: The University of Chicago Press. 1960/1980.
22. Thompson N.A., Weiss D.J. A framework for the development of computerized adaptive tests // *Practical Assessment, Research & Evaluation*. 2011. № 16(1). P. 19.
23. de la Torre J., Patz R.J. Making the Most of What We Have: A Practical Application of Multidimensional Item Response Theory in Test Scoring // *Journal of Educational and Behavioral Statistics*. 2005. № 30(3). P. 295311. doi:10.3102/10769986030003295.
24. Wright B.D., Masters G.N. Rating scale analysis. Rasch measurements // Chicago: MESA Press. 1982. 206 p.



Model of Adaptive Learning and His Implementation

Denis A. Pominov*

Moscow State University of Psychology and Education, Moscow, Russia,
ORCID: <https://orcid.org/0000-0002-1321-3713>
e-mail: pominovda@mgppu.ru

The presented project is aimed at automating the e-learning process regarding the acquisition of practical skills for solving non-formalized tasks, determining the level of knowledge and reducing the duration of training by reducing the number of tasks depending on the level of training. To meet these requirements, an adaptive testing approach was implemented and “Adaptive trainer” web-service was implemented to demonstrate how it works.

Keywords: adaptive training, markov random process, adaptive trainer, self-learning systems, information systems implementation.

For citation:

Pominov D.A. Model of Adaptive Learning and His Implementation. *Modelirovanie i analiz dannyykh = Modelling and Data Analysis*, 2020. Vol. 10, no. 3, pp. 39–52 . DOI: <https://doi.org/10.17759/mda.2020100303> (In Russ., abstr. in Engl.).

Funding. This work has been funded by the Moscow State University of Psychology and Education.

References

1. Kuravsky L.S., Artemenkov S.L., Yuriev G.A., Grigorenko E.L. New approach to computer-based adaptive testing. – *Experimental Psychology*. 2017. Vol. 10. No. 3. Pp. 33–45. doi:10.17759/expsy.2017100303 (in Russian).
2. Kuravsky L.S., Kulik S.D., Marmalyuk P.A., Yuriev G.A. Certificate of state registration of the useful model № 118095 A device for modeling adaptive testing of cognitive abilities of the subject. – Application № 2012105993/08; decrealed 21 February 2012; registered 10 July 2012. – (ROSPATENT).
3. Kuravsky L.S., Margolis A.A., Marmalyuk P.A., Yuriev G.A., Dumin P.N. Trainable Markov models in tasks of optimization of the order of presentation of psychological tests // *Neurocomputers: Development and Application*. 2013. No. 4. pp. 28–38 (in Russian).
4. Kuravsky L.S., Margolis A.A., Marmalyuk P.A., Panfilova A.S. , Yuriev G.A. Mathematical aspects of the adaptive trainer concept – *Psychological Science and Education*. 2016. Vol. 21. No. 2. Pp. 84–95. doi: 10.17759/pse.2016210210 (in Russian).
5. Kuravsky L.S., Margolis A.A., Yuriev G.A., Pominov D.A. Concept of self-learning adaptive trainer // *Neurocomputers: Development and Application*. 2018. No 3. pp. 29–37 (in Russian).

***Denis A. Pominov**, Research Scholar, Computer Science Faculty, Moscow State University of Psychology and Education, Russia, ORCID: <https://orcid.org/0000-0002-1321-3713>, e-mail: pominovda@mgppu.ru



6. Kuravsky L.S., Margolis A.A., Marmalyuk P.A., Yuriev G.A., Dumin P.N., Kulik S.D. Certificate of state registration of the useful model № 122796 Decision support system for psychological and pedagogical testing. – Application № 2012132684/08; decrealed 31 July 2012; registered 10 December 2012. – (ROSPATENT).
7. Kuravsky L.S., Margolis A.A., Yuriev G.A., Marmalyuk P.A. Concept of a system of support in decision making for psychological testing. – Psychological science and education, No. 1, 2012, pp. 56–65 (in Russian).
8. Kuravsky L.S., Yuriev G.A. Adaptive testing as a Markovian process: models and their identification. – Neurocomputers: Development and Application, No. 2, 2011, pp. 21–29 (in Russian).
9. Kuravsky L.S., Yuriev G.A. Probabilistic method of filtering artefacts in adaptive testing. – Experimental Psychology, Vol.5, No. 1, 2012, pp. 119–131 (in Russian).
10. Kuravsky L.S., Yuriev G.A. Use of Markov models in processing testing results. – Psychology Issues, No. 2, 2011, pp. 98–107 (in Russian).
11. Kuravsky L.S., Yuriev G.A. About an approach to adaptive testing and eliminating its artefacts. // Neurocomputers: Development and Application, No. 1, 2012 (in Russian).
12. Markov models in diagnostics and forecasting tasks: Study Guide. // Edited by L.S. Kuravsky. – 2nd edition, enlarged -Moscow: Publishing House of the Moscow State University of Psychology and Education, 2017. 203 p. (in Russian).
13. 1C: E-learning [Web resource] // URL: <http://v8.1c.ru/elo> (In Russian).
14. Baker F.B. The Basics of Item Response Theory // ERIC Clearinghouse on Assessment and Evaluation. University of Maryland, College Park. MD. 2001.
15. Edmodo [Web resource] // Connect with students and parents in your paperless classroom. URL: <https://www.edmodo.com>.
16. Gregory R.J. Psychological testing: History, principles, and applications (5th edition). New York: Pearson. 2007.
17. Kats Y. Learning Management Systems and Instructional Design: Best Practices in Online Education. IGI Global. 2013. ISBN 9781466639317
18. Kuravsky L.S., Marmalyuk P.A., Yuryev G.A., Dumin P.N. A Numerical Technique for the Identification of Discrete-State Continuous-Time Markov Models // Applied Mathematical Sciences. 2015. V. 9. № 8. P. 379–391. URL: <http://dx.doi.org/10.12988/ams.2015.410882>.
19. Kuravsky L.S., Marmalyuk P.A., Baranov S.N., Alkhimov V.I., Yuryev G.A., Artyukhina S.V. A New Technique for Testing Professional Skills and Competencies and Examples of its Practical Applications // Applied Mathematical Sciences. 2015. V. 9. № 21. P. 1003–1026. <http://dx.doi.org/10.12988/ams.2015.411899>.
20. Moodle open-source learning platform [Web resource] // Moodle Pty Ltd. URL:<https://moodle.org> (дата обращения 03.02.2016).
21. Rasch G. Probabilistic models for some intelligence and attainment tests // Copenhagen, Danish Institute for Educational Research, expanded edition (1980) with foreword and afterword by B.D. Wright. Chicago: The University of Chicago Press. 1960/1980.
22. Thompson N.A., Weiss D.J. A framework for the development of computerized adaptive tests // Practical Assessment, Research & Evaluation. 2011. № 16 (1). P. 19.
23. de la Torre J., Patz R.J. Making the Most of What We Have: A Practical Application of Multidimensional Item Response Theory in Test Scoring // Journal of Educational and Behavioral Statistics. 2005. № 30(3). P. 295311. doi:10.3102/10769986030003295.
24. Wright B.D., Masters G.N. Rating scale analysis. Rasch measurements // Chicago: MESA Press. 1982. 206 p.

◇◇◇КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ◇◇◇

УДК 004.921

Исследование и разработка методов моделирования и анимации трехмерных объектов

Чернышов Л.Н.*

Московский авиационный институт (МАИ),
г. Москва, Российская Федерация
ORCID: <https://orcid.org/0000-0002-1512-4052>
e-mail: levchern@gmail.com

Захарова Е.Д.**

Московский авиационный институт (МАИ),
г. Москва, Российская Федерация
ORCID: <https://orcid.org/0000-0002-9220-0514>
e-mail: eugenezaharova@gmail.com

В статье рассматриваются уже известные методы моделирования трехмерных объектов с их последующей анимацией. Предлагается комбинированный метод моделирования, включающий в себя особенности известных методов, который сокращает время проектирования модели, сохраняя при этом ее реалистичность.

Ключевые слова: моделирование, анимация, методы моделирования, трехмерные объекты, реалистичность.

Для цитаты:

Чернышов Л.Н., Захарова Е.Д. Исследование и разработка методов моделирования и анимации трехмерных объектов // Моделирование и анализ данных. 2020. Том 10. № 3. С. 53–59. DOI: <https://doi.org/10.17759/mda.2020100304>

ВВЕДЕНИЕ

Актуальность реалистичности компьютерной графики в последние годы заметно возросла. Практически на каждом шагу мы встречаемся с ее проявлениями, от филь-

**Чернышов Лев Николаевич*, кандидат физико-математических наук, доцент кафедры вычислительной математики и программирования, Московский авиационный институт (МАИ), Москва, Россия, ORCID: <https://orcid.org/0000-0002-1512-4052>, e-mail: levchern@gmail.com

***Захарова Евгения Дмитриевна*, студентка 2 курса магистратуры, Московский авиационный институт (МАИ), Москва, Россия, ORCID: <https://orcid.org/0000-0002-9220-0514>, e-mail: eugenezaharova@gmail.com



мов и компьютерных игр до GPS-навигаторов и проектирования техники. Однако реалистичность в вопросах компьютерной графики по-прежнему остается камнем преткновения, а любое неестественное движение персонажа в фильме или неточность в механизме вызывает шквал критики в адрес создателей и аниматоров.

Вопросам исследования методов моделирования посвящено много работ [1, 2, 3, 4], но, как можно видеть в повседневной жизни, проблема реалистичной графики остается не просто живой и актуальной – она является одной из самых острых задач из когда-либо поставленных перед программистами.

Реалистичность компьютерных трехмерных моделей и их анимация в последнее время вышла на новый уровень, особенно в тех случаях, когда моделирование касается объектов живой природы. Но здесь возникает проблема реализации таких моделей – на подобную работу уходит очень много времени и дополнительных затрат, а также требует глубоких знаний анатомии моделируемых объектов. С теми же трудностями сталкиваются специалисты, занимающиеся моделированием технических объектов.

Работа посвящена сравнению уже существующих методов моделирования трехмерных объектов, их эффективности и реалистичности результата. Целью исследования является разработка нового метода моделирования, позволяющего сократить время проектирования и уменьшить сложность создания модели, сохранив при этом реалистичность результата.

СРАВНЕНИЕ МЕТОДОВ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ И АНИМАЦИИ

На данный момент среди наиболее используемых методов компьютерного моделирования можно выделить четыре. Реалистичность результатов каждого метода варьируется, как и время проектирования модели. Рассмотрим каждый из методов чуть подробнее.

Технология блендшейпов предполагает работу с человеческими моделями и направлена на реалистичное воспроизведение черт лица. Создается несколько крайних точек и в них фиксируется то или иное выражение лица, например, улыбка или удивление.

Скелетная анимация позволяет создать модель объекта с нуля. Проектируется каркас объекта – скелет – в виде древообразной структуры костей. Каждая кость связана с предыдущей, а каждая вершина – с какой-либо костью. При движении отдельной кости двигаются все вершины, что создает довольно реалистичную анимацию. Именно этот способ чаще других применяется в мультипликации.

Технология захвата движений активно используется в кинематографе и предусматривает размещение специальных датчиков на костюме человека, который затем имитирует движение того или иного объекта, а датчики фиксируют его положение.

Система мышц, пожалуй, наиболее эффективный и наиболее сложный метод компьютерного моделирования. В процессе проектирования к скелету модели в соответствии с анатомическими особенностями прикрепляются так называемые мышцы, которые имитируют реальную мышечную массу, после чего им задается динамическое поведение. В итоге получается реалистичная картинка.



В ходе исследования было сделано подробное сравнение перечисленных методов компьютерного моделирования, проведен анализ каждого метода с учетом затрачиваемого времени проектирования, сложности модели, степени реалистичности и дополнительных затрат. Ниже представлена таблица 1 с результатами сравнения.

Таблица 1

Сравнительный анализ методов компьютерного моделирования

	Блендшейпы	Скелетная анимация	Захват движений	Система мышц
Временные затраты	средне	долго	средне	долго
Сложность проектирования	средне	сложно	легко	сложно
Сложность при создании реалистичности	сложно	сложно	легко	средне
Возможность внесения изменений	нет	да	нет	да
Дополнительные затраты	нет	нет	да	нет
Дополнительные знания	нет	нет	да	да

В результате сравнения было выявлено, что наиболее эффективным из перечисленных методов стала система мышц – но она же является довольно сложной технологией и занимает много времени. Также стоит отметить, что не каждый специалист сможет ей воспользоваться без предварительной подготовки и должных знаний в анатомии.

ХОД ИССЛЕДОВАНИЯ

После сравнительного анализа уже существующих методов компьютерного моделирования возник вопрос разработки такого метода, который будет менее сложен в использовании, займет меньше времени на проектирование модели, но в результате даст такую же реалистичную модель. Были более подробно изучены скелетная анимация и система мышц, и в результате решение оказалось очень простым – комбинированный метод, включающий в себя особенности скелетной анимации и системы мышц.

Если рассматривать стандартные методы использования системы мышц, можно заметить, что огромная доля времени уходит на создание полного полигонального скелета объекта и прикрепление мышц к нему. Однако можно отказаться от полигонального скелета и крепить мышцы непосредственно к скелетной системе управления – небольшому скелетному каркасу, отвечающему за большую часть движений объекта, – а также сократить количество мышц до минимума, оставив только те, что будут четче всего выражены при анимации.



Таким образом, весь алгоритм проектирования модели по комбинированному методу сводится к следующим шагам:

1. Создание первоначального эскиза модели. Подробное описание модели в различном ракурсе, изучение общей анатомии модели.
2. Проектирование по эскизу трехмерной модели без скелетной системы управления, мышц и текстур. Наложение полигональной сетки для дальнейшей анимации модели.
3. Создание скелетной системы управления. Основной каркас модели без подробных анатомических деталей, отвечающий за ее движения. Скелетную систему управления лучше всего создавать, если модель – особенно в случае объекта живой природы – находится в нейтральной спокойной позе.
4. Распределение весов взаимодействия полученного скелета с кожным покровом. Указывается степень воздействия костей на ту или иную вершину модели для максимально реалистичного результата.
5. Прикрепление ярко выраженных мышц к скелетной системе управления.
6. Распределение весов взаимодействия мышц с кожным покровом.
7. Наложение текстуры. Создание кожного покрова модели.

Такой подход существенно сокращает время моделирования и уменьшает объем работы, при этом результат по-прежнему остается достаточно реалистичным и не требует от разработчика углубленных знаний анатомии. Преимущества и недостатки разработанного метода перечислены в таблице 2.

Таблица 2

Преимущества и недостатки комбинированного метода компьютерного моделирования

Преимущества	Недостатки
Сокращение затрат по времени	Средняя реалистичность по сравнению с классической схемой работы с системой мышц
Уменьшение объектов моделирования по сравнению с классической схемой работы с системой мышц	Создание только самых необходимых мышц влечет отсутствие эффекта скольжения
Углубленные знания анатомии не требуются	
Отсутствие необходимости создания дополнительных полигональных объектов	

АПРОБАЦИЯ И РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ

Предложенный метод был использован при создании моделей в программе Maya. Было создано две модели – одна по стандартному методу скелетной анимации, вторая по комбинированному методу.

В случае с первой моделью время разработки заняло 56 часов. Был настроен полный полигональный скелет объекта, создан каркас модели, применены текстуры.

Создание второй модели заняло 40 часов. Была построена скелетная система управления в виде основного скелетного каркаса, отвечающего за движение модели. Затем к нему были прикреплены мышцы, визуально наиболее заметные при движении модели, после чего также были применены текстуры. На рисунке 1 представлены результаты проектирования обеих моделей.

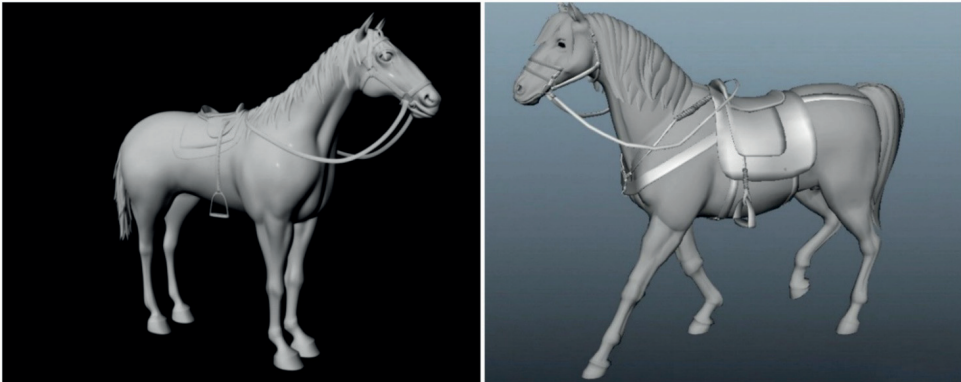


Рис. 1. Результаты проектирования моделей по методу скелетной анимации (слева) и комбинированному методу (справа)

Как можно видеть, реалистичность второй модели не уступает первой, но при этом время создания моделей существенно отличается. Больше всего времени при работе с первой моделью было уделено созданию полного скелета объекта, хотя большая часть костей не участвует в движении. Кроме того, было трудно воссоздать полный скелет в соответствии с анатомией объекта. При проектировании второй модели таких сложностей не возникло.

ЗАКЛЮЧЕНИЕ

Исследование моделирования трехмерных объектов является интересной и актуальной задачей. Для сравнительного анализа существующих методов моделирования была подробно изучена предметная область и проанализированы наиболее эффективные методы моделирования. Анализ показал, что практически все используемые методы либо слишком сложны в проектировании, либо слишком затратны по времени.

Был предложен комбинированный метод моделирования, включающий в себя особенности наиболее эффективных методов моделирования – скелетной анимации и системы мышц. В процессе апробации было выявлено, что комбинированный метод занимает меньше времени при создании моделей, он менее сложен в использовании, а полученная модель обладает достаточной реалистичностью.

Предложенный метод можно использовать не только в мультипликации и кинематографе. С его помощью можно спроектировать краш-тесты в областях автомобилестроения для предотвращения аварийных ситуаций, спрогнозировать состояние



пилота во время полета или определить степень безопасности отдельных инженерных разработок. Кроме того, подобная техника моделирования позволит подготовить большее количество специалистов, результаты работы которых будут не хуже работы сегодняшних профессионалов, которую мы видим в повседневной жизни.

Литература

1. Флеминг Б., Доббс Д. Методы анимации лица. Мимика и артикуляция [пер. с англ.]; под общ. ред. Козицкой Е.А. – М. : ДМК Пресс, 2002. – 333 с.
2. Мараффи К. Создание персонажей в Maya: моделирование и анимация [пер. с англ. и ред. В.А. Коваленко]. – М. : ООО И.Д. Вильямс, 2004. – 441 с.
3. Кундерт-Гиббс Д., Ларкинс М., Деракшани Д., Кунзендорф Э. Освоение Maya 8.5 [Текст]. – М. : ООО И.Д. Вильямс, 2007. – 928 с.
4. Lewis J.P., Cordner M., Fong N. Proceedings of the 27th annual conference on Computer graphics and interactive techniques. – SIGGRAPH '00. – New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 2000. – pp. 165–172.



Research and Development of Simulation Methods and Animation of Three-Dimensional Objects

Lev N. Chernyshov*

Moscow Aviation Institute, Moscow, Russia,
ORCID: <https://orcid.org/0000-0002-1512-4052>
e-mail: levchern@gmail.com

Eugene D. Zakharova**

Moscow Aviation Institute, Moscow, Russia,
ORCID: <https://orcid.org/0000-0002-9220-0514>
e-mail: eugenezaharova@gmail.com

The article discusses already known methods of modeling three-dimensional objects with their subsequent animation. A combined modeling method is proposed, which includes the features of known methods, which reduces the design time of the model, while maintaining its realism.

Keywords: modeling, animation, methods of modeling, three-dimensional objects, realism.

For citation:

Chernyshov L.N., Zakharova E.D. Research and Development of Simulation Methods and Animation of Three-Dimensional Objects. *Modelirovanie i analiz dannykh = Modelling and Data Analysis*, 2020. Vol. 10, no. 3, pp. 53–59. DOI: <https://doi.org/10.17759/mda.2020100304> (In Russ.).

References

1. Fleming B., Dobbs D. Animating Facial Features and Expressions, 2002. (In Russ.).
2. Maraffi C. Maya Character Creation: Modeling and Animation Controls, 2003. (In Russ.).
3. Kundert-Gibbs J., Larkins M., Derakhshani D., Kunzendorf E. Mastering Maya 8.5. *Wiley*, 2007. (In Russ.).
4. Lewis J.P., Corder M., Fong N. Proceedings of the 27th annual conference on Computer graphics and interactive techniques. – SIGGRAPH '00. – New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 2000. – pp. 165–172.

***Lev N. Chernyshov**, Candidate of Physical and Mathematical Sciences, Associate Professor, Department of Computational Mathematics and Programming, Moscow Aviation Institute, Moscow, Russia, ORCID: <https://orcid.org/0000-0002-1512-4052>, e-mail: levchern@gmail.com

****Eugene D. Zakharova**, 2nd year MSc student, Moscow Aviation Institute, Moscow, Russia, ORCID: <https://orcid.org/0000-0002-9220-0514>, e-mail: eugenezaharova@gmail.com

◇◇◇◇◇◇◇◇◇◇ МЕТОДЫ ОПТИМИЗАЦИИ ◇◇◇◇◇◇◇◇◇◇

УДК 519.85

Практическая реализация алгоритма декомпозиции путей ориентированного графа

Золотарев И.А. *

Московский авиационный институт (НИУ МАИ),
г. Москва, Российская Федерация
ORCID: <https://orcid.org/0000-0002-6437-2212>
e-mail: yngvar.antonsson@gmail.com

Рассказова В.А. **

Московский авиационный институт (НИУ МАИ),
г. Москва, Российская Федерация,
ORCID: <https://orcid.org/0000-0003-4943-3133>
e-mail: varvara.rasskazova@mail.ru

Работа направлена на прояснение некоторых особенностей программной реализации алгоритма декомпозиции путей ориентированного графа. Разобраны алгоритмы для формирования таблицы M для декомпозиции множества путей, сортировки таблицы M по полю N_x и расчета балансов. На основе данных алгоритмов и исходного алгоритма декомпозиции путей ориентированного графа разработан комплекс программ на языке программирования Python. Проведены расчеты для случайного графа размерности 100 вершин и приводится время работы предложенного алгоритма. Полученные результаты могут быть использованы при решении задачи организации грузовых железнодорожных перевозок на этапе назначения и перемещения локомотивов. Научная и практическая новизна работы заключается в существенном снижении размерности исходной задачи, что особенно важно в условиях транспортных сетей сложной топологии.

Ключевые слова: теория оптимизации, оптимизация на графах, алгоритм декомпозиции путей ориентированного графа, сильно связный граф, подграфы.

Для цитаты:

Золотарев И.А., Рассказова В.А. Практическая реализация алгоритма декомпозиции путей ориентированного графа // Моделирование и анализ данных. 2020. Том 10. № 3. С. 60–68. DOI: <https://doi.org/10.17759/mda.2020100305>

*Золотарев Игорь Антонович, студент магистратуры, Московский авиационный институт, (НИУ МАИ), г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0002-6437-2212>, e-mail: yngvar.antonsson@gmail.com

**Рассказова Варвара Андреевна, кандидат физико-математических наук, доцент кафедры 804 «Теория вероятностей и компьютерное моделирование», Московский авиационный институт, (НИУ МАИ), г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0003-4943-3133>, e-mail: varvara.rasskazova@mail.ru



ВВЕДЕНИЕ

Практическая реализация алгоритма декомпозиции путей ориентированного графа вызывает несколько побочных задач, требующих решения. Во-первых, нужно построить таблицу, которая будет использоваться в алгоритме. Ручное составление таблицы и ручная обработка данных требует колоссальных временных затрат, особенно для графов большой размерности, пути в которых, как правило, могут достигать нескольких сотен (и даже тысяч) ребер (вершин). Таким образом, совершенно очевидна необходимость частичной автоматизации вышеописанного процесса. Составление таблицы по исходному графу и множеству путей – далеко не тривиальная проблема, рассмотрение которой выходит за пределы данной статьи, поэтому будем полагать, что на вход алгоритма подаются набор путей, а также таблица, в которой указана принадлежность ребер графа к определенным подграфам в виде списка.

Второй задачей, требующей решения в процессе реализации алгоритма, является использование в алгоритме нескольких эвристических критериев, таких как расчет балансов и сортировка по N_{Σ} .

ПОДГОТОВКА ТАБЛИЦЫ ДЛЯ ДЕКОМПОЗИЦИИ

Пусть задан ориентированный граф $\vec{G} = (V, E)$ и набор подграфов $G_1, G_2 \dots G_k$, и пусть $P = \{p_1, \dots, p_m\}$ – заданный набор путей графа \vec{G} . Будем полагать, что исходные данные подаются на вход алгоритма в следующем виде: $paths = \{(v1, v2, v3, \dots), \{(v1, v2, v3, \dots), \dots\}$ – список путей, v_i – числовые значения вершины i в пути j , $i = 1..m, j = \in \{1, 2, \dots, |V|\}$, $E_{subgraph}(v_i, v_j) = \{l | (v_i, v_j) \in G_l\}$ – список принадлежности ребер графа подграфам.

На следующем этапе нужно составить таблицы $M(p_i)$, которые затем будут объединены в таблицу $M(p)$. Для этого необходимо преобразовать каждый из путей вида $(v1, v2, v3, \dots)$ для пути p_i в набор $(v_{1j}, s_{1j}, v_{2j}, s_{2j}, v_{3j}, \dots, v_{l-1j}, s_{l-1j}, v_{lj})$, $s_{kj} \in E_{subgraph}(v_{kj}, v_{k+1j})$ множество которых $path_{mult}$ содержит все возможные варианты прохождения по подграфам в этом пути. Опишем простой алгоритм для данного действия.

Алгоритм 1. Генерация вариантов прохождения по подграфам в пути

1. Пусть имеется путь $p = (v1, v2, v2, \dots)$ и таблица $E_{subgraph}$, описанная выше. Определим пустую таблицу $path_{mult} = ()$
2. Помещаем $v1$ в таблицу: $path_{mult} = (v1)$. Если $E_{subgraph}(v1, v2) = \{s1\}$, то помещаем $s1$ в таблицу: $path_{mult} = (v1, s2)$. Если $E_{subgraph}(v1, s2) = \{s1, s1_2, \dots, s1_m\}$, тогда составляем таблицу следующего вида:

$$\begin{pmatrix} v1 & s1 \\ v1 & s1_2 \\ \dots & \dots \\ v1 & s1_m \end{pmatrix}$$



3. Помещаем v_2 в таблицу путем добавления ее справа к каждой строке таблицы. Продолжаем добавлять вершины и числа принадлежности ребра к подграфу до тех пор, пока не встретится $|E_{subgraph}(v_i, v_{i+1})| = \{s_{i_1}, s_{i_2}, \dots, s_{i_z}\} > 1$
4. Пусть на i -м шаге имеется таблица следующего вида:

$$\begin{pmatrix} v_1 & s_{1_1} & v_2 & s_2 & v_3 & \dots & v_i \\ v_1 & s_{1_2} & v_2 & s_2 & v_3 & \dots & v_i \\ & & \dots & & & & \\ v_1 & s_{1_m} & v_2 & s_2 & v_3 & \dots & v_i \end{pmatrix}$$

Тогда добавляем значения $\{s_{i_1}, s_{i_2}, \dots, s_{i_z}\}$ следующим образом:

Далее переходим к шагу 2, с вершиной v_{i+1} вместо v_1 .

$$\begin{pmatrix} v_1 & s_{1_1} & v_2 & s_2 & v_3 & \dots & v_i & s_{i_1} \\ v_1 & s_{1_2} & v_2 & s_2 & v_3 & \dots & v_i & s_{i_1} \\ & & \dots & & & & & \\ v_1 & s_{1_m} & v_2 & s_2 & v_3 & \dots & v_i & s_{i_1} \\ v_1 & s_{1_1} & v_2 & s_2 & v_3 & \dots & v_i & s_{i_2} \\ v_1 & s_{1_2} & v_2 & s_2 & v_3 & \dots & v_i & s_{i_2} \\ & & \dots & & & & & \\ v_1 & s_{1_m} & v_2 & s_2 & v_3 & \dots & v_i & s_{i_{z-1}} \\ v_1 & s_{1_1} & v_2 & s_2 & v_3 & \dots & v_i & s_{i_z} \\ v_1 & s_{1_2} & v_2 & s_2 & v_3 & \dots & v_i & s_{i_z} \\ & & \dots & & & & & \\ v_1 & s_{1_m} & v_2 & s_2 & v_3 & \dots & v_i & s_{i_z} \end{pmatrix}$$

5. Алгоритм продолжается до тех пор, пока не закончатся вершины в пути.

Далее работа сводится к простым операциям конкатенации таблиц и заполнения их, следуя описанным в оригинальной статье принципам.

ДЕКОМПОЗИЦИЯ

Перейдем непосредственно к алгоритму декомпозиции. Здесь и далее будут использоваться обозначения из [1]. Как было упомянуто выше, здесь есть два момента, связанных с эвристиками локального поиска.

Во-первых, это сортировка по N_Σ . Чтобы ускорить обработку данных, сортировку по N_Σ можно не проводить, если выполняются следующие условия:

1. после сортировки по столбцам mk , c и bal_Σ верхняя строка таблицы M , где в столбце mk находится «1», единственная;
2. после сортировки по столбцам mk , c и bal_Σ верхние строки таблицы M , где в столбце mk находится «1» для одного и того же p содержат хотя бы два различных значения в столбце c ;
3. после сортировки по столбцам mk , c и bal_Σ верхние строки таблицы M , где в столбце mk находится «1» для одного и того же p содержат хотя бы два разных значения в столбце bal_Σ .



Если ни одно из вышеперечисленных условий не выполняется, расчет N_z организован следующим образом:

1. для каждой строки из множества $M(p_i)$ добавляем ее во множество $M(D)$ – множество отобранных в декомпозицию строк на i -м шаге;
2. рассчитываем N_z ;
3. для невыбранных на i -м этапе строк значения N_z можно не рассчитывать, так как после сортировки они всё равно окажутся «ниже», чем значения из $M(p_i)$. Для определенности можно считать, что они равны ∞ .

Второй эвристический алгоритм – это расчет баланса. Перепишем расчет баланса в вид алгоритма, который можно будет легко перевести в программный код:

Алгоритм 2. Расчет баланса

1. Пусть имеется таблица M после выбора строки в декомпозицию, nom – строка таблицы, выбранная на предыдущем шаге, $p(nom)=p_i$, $Ind(nom)=Ind_i$.
2. Для каждого s из $[1..K]$ заведем массив для условия отбора строк $Cond(s)$ с размерностью $|M| \times 1$, где $|M|$ – количество строк в таблице M .
3. Для каждого s из $[1..K]$ и для каждого i из $[1..I-1]$ посчитаем условие следующего вида $cond_i(z) = \text{Истина}$, если в z -й строке таблицы M : $vi = vi(nom)$ и $vi+1 = vi+1(nom)$ и $si = s$. Далее, выполняется построчное ИЛИ для строк из $cond_i$ и $Cond$.
4. Для каждого s из $[1..K]$ отбираем строки из таблицы M по условию: $M_z[p] \neq p_i$ И $M_z[mk] = 1$ И $M_z[bal_s] = 0$ И $M_z[Ind] \neq Ind_i$ И $Cond(s)$ и выставляем значение в столбце bal_s равным 1 для отобранных строк.

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

В качестве языка программирования был выбран Python 3. Этот язык обладает простым синтаксисом, удобными для работы типами данных, а также для него были реализованы математические пакеты с хорошим API (numpy) и мощное средство для работы с таблицами (pandas). Функции из этих пакетов лягут в основу программного кода для реализации алгоритма декомпозиции путей ориентированного графа. Кроме этого, будет использована библиотека graphviz для визуализации графа. При реализации будет использован объектно-ориентированный подход. Он позволит объединить в одном месте данные и код для работы с ними, а также скрыть детали реализации алгоритма от пользователей (принцип инкапсуляции). Входные данные будут подаваться через конструктор, в котором будет происходить создание исходной таблицы, а единственным, кроме конструктора, открытым методом будет метод, в котором происходит непосредственно декомпозиция.

Для тестирования практической реализации алгоритма декомпозиции необходимо сгенерировать сильно связный ориентированный граф и множество путей.

Пусть имеется граф \vec{G} , изображенный на рис. 1 и 2.

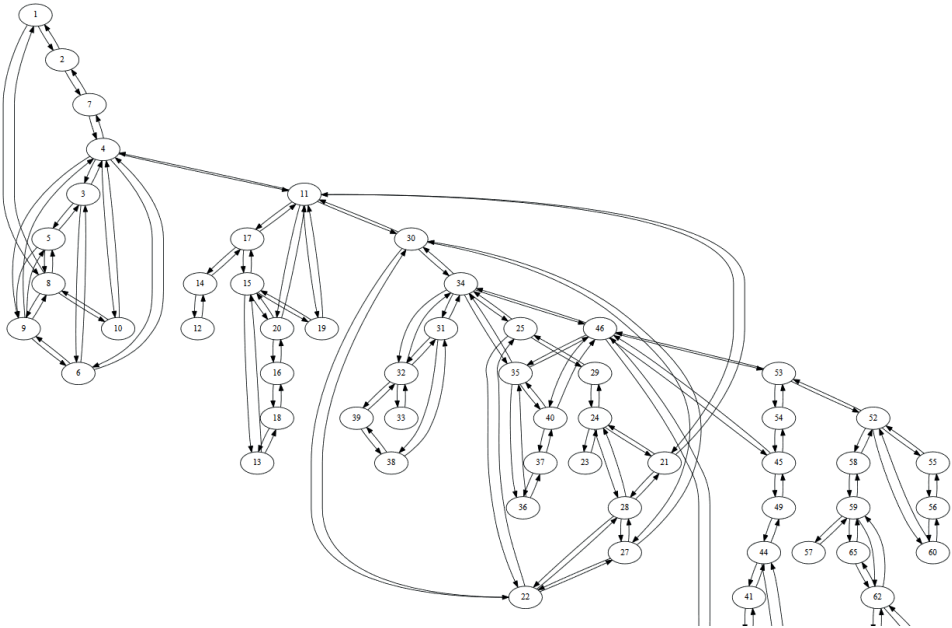


Рис. 1. Сильно связный граф (1)

Выберем множество путей

$$P = \{$$

$$p_1 = \{1, 2, 7, 4, 11, 30, 22, 25, 34, 46, 53, 52, 58, 59, 65, 62, 70, 67\},$$

$$p_2 = \{11, 17, 15, 13, 18, 16, 20, 11, 30, 27, 22, 30, 34, 46\},$$

$$p_3 = \{38, 31, 34, 30, 11, 21, 28, 22, 30\},$$

$$p_4 = \{1, 2, 7, 4, 3, 5, 8, 9, 6\},$$

$$p_5 = \{5, 3, 4, 7, 2, 1\}$$

$$\}$$

Будем считать, что граф состоит из 10 подграфов, G_1, \dots, G_{10}

$$V(G_1) = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 30\}$$

$$V(G_2) = \{11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 30\}$$

$$V(G_3) = \{22, 30, 31, 32, 33, 34, 38, 39, 46\}$$

$$V(G_4) = \{11, 21, 22, 23, 24, 25, 27, 28, 29, 30, 34, 35, 36, 37, 40, 46\}$$

$$V(G_5) = \{41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 53, 54\}$$

$$V(G_6) = \{52, 53, 54, 55, 56, 57, 58, 59, 60, 62, 65\}$$

$$V(G_7) = \{61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 75\}$$

$$V(G_8) = \{66, 68, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 85\}$$

$$V(G_9) = \{81, 82, 83, 84, 85, 86, 87, 88, 89, 90\}$$

$$V(G_{10}) = \{85, 88, 91, 93, 94, 95, 96, 97, 96, 97, 98, 99, 100\}$$

$$E(G_l) = \{(v_i, v_j) | v_i, v_j \in G_l\}, l \in \{1, \dots, 10\}$$

Для G_1, \dots, G_{10} , $E(G_l) = \{(v_i, v_j) | v_i, v_j \in G_l\}$, $l \in \{1, \dots, 10\}$



Рис. 2. Сильно связный граф (2)

Пути из P лежат в G_1, \dots, G_7 .

Составим таблицу $E_{subgraph}$, которая используется в алгоритме 1. После выполнения алгоритмов 1 и 2 и алгоритма, описанного в [1] получаем, что пути находятся в следующих подграфах:

$$p_1 \in \{G_1\}, p_2 \in \{G_3, G_4\}, p_3 \in \{G_2, G_3\}, p_4 \in \{G_3, G_5, G_6, G_7\}, p_5 \in \{G_1\}$$

Работа алгоритма для графа выше занимает в среднем 6 секунд.

Результаты запусков алгоритма для разных входных данных приведены в табл. 1.



Таблица 1

Результаты запусков алгоритма декомпозиции

Количество вершин	Количество ребер	Количество путей	Количество подграфов	Время работы алгоритма
6	5	1	1	20 мс
6	14	4	3	800 мс
16	28	2	3	1.2 с
27	36	2	7	1.6 с
70	130	5	7	6 с
100	160	7	10	11 с

ЗАКЛЮЧЕНИЕ

Разработаны алгоритмы для подготовки таблицы для декомпозиции и подробнее описаны алгоритмы декомпозиции. Полученные результаты реализованы в виде программного кода и проверены для ориентированного графа размерности 100 (вершин).

Литература

1. Гайнанов Д.Н., Кобылин А.В., Рассказова В.А. Моделирование грузовых железнодорожных перевозок методами теории графов и комбинаторной оптимизации // Автоматика и телемеханика. 2016. № 11. С. 60–79.
2. Гайнанов Д.Н., Кибзун А.И., Рассказова В.А. Теоретико-графовый алгоритм решения задачи о назначении и перемещении локомотивов // Вестник компьютерных и информационных технологий. 2017. № 5. С. 51–56.
3. Гайнанов Д.Н., Кибзун А.И., Рассказова В.А. Задача о декомпозиции множества путей ориентированного графа и ее приложение // Автоматика и телемеханика. 2018. № 12. С. 142–166.



Practical Realization of Algorithm of Oriented Graph Paths Decomposition

Igor A. Zolotarev*

Moscow Aviation Institute, Moscow, Russia,
ORCID: <https://orcid.org/0000-0002-6437-2212>
e-mail: yngvar.antonsson@gmail.com

Varvara A. Rasskazova**

Moscow Aviation Institute, Moscow, Russia,
ORCID: <https://orcid.org/0000-0003-4943-3133>
e-mail: varvara.rasskazova@mail.ru

This study aims to clarify the methodological status program realization of oriented graph paths decomposition. Algorithms of matrix formulation of decomposition table M from multitude of paths, table sorting by N_y and balance computing explored. On the basis of those algorithms and original algorithm of oriented graph paths decomposition realized Python 3 program. Results for random-generated graph size of 100 vertices computed and time measured. The results obtained can be used to solve the problem of organizing freight rail transportation at the stage of assignment and movement of locomotives. The scientific and practical novelty of the work lies in a significant reduction in the dimension of the original problem, which is especially important in the conditions of transport networks of complex topology.

Keywords: optimization theory, graph optimization, algorithm of directed graph paths decomposition, strongly connected graph, subgraphs.

For citation:

Zolotarev I.A., Rasskazova V.A. Practical Realization of Algorithm of Oriented Graph Paths Decomposition. *Modelirovanie i analiz dannykh = Modelling and Data Analysis*, 2020. Vol. 10, no. 3, pp. 60–68. DOI: <https://doi.org/10.17759/mda.2020100305> (In Russ., abstr. in Engl.).

References

1. Gainanov D.N., Konygin A.V., Rasskazova V.A. Modelling railway freight traffic using the methods of graph theory and combinatorial optimization. *Automation and Remote Control*, 2016 vol. 77, no. 11, pp. 1928–1943.
2. Gainanov D.N., Kibzun A.I., Rasskazova V.A. Theoretical-graph algorithm in the problem on the assignments and transportations of locomotives. *Herald of computer and information technologies*, 2017, no. 5, pp. 51–56.

***Igor A. Zolotarev**, Master Student, Moscow Aviation Institute, Moscow, Russia, ORCID: <https://orcid.org/0000-0002-6437-2212>, e-mail: yngvar.antonsson@gmail.com

****Varvara A. Rasskazova**, PhD in Physics and Mathematics, Associate Professor, Moscow Aviation Institute, Moscow, Russia, ORCID: <https://orcid.org/0000-0003-4943-3133>, e-mail: varvara.rasskazova@mail.ru



3. Gainanov D.N., Kibzun A.I., Rasskazova V.A. The Decomposition Problem for the Set of Paths in a Directed Graph and Its Application. *Automation and Remote Control*, 2018, vol. 79, no. 12, pp. 2217–2236.

Статистическое моделирование ядра вероятностного распределения и его применение к решению задачи квантильной оптимизации с билинейной функцией потерь

Ардабьевский П.А.*

Московский авиационный институт
(национальный исследовательский университет),
г. Москва, Российская Федерация
ORCID: <https://orcid.org/0000-0002-6857-7485>
e-mail: ardpavel@gmail.com

Гончар Д.А.**

Московский авиационный институт
(национальный исследовательский университет),
г. Москва, Российская Федерация
ORCID: <https://orcid.org/0000-0002-2414-491X>
e-mail: dima.gonchar.29.08.13@gmail.com

Кан Ю.С.***

Московский авиационный институт
(национальный исследовательский университет),
г. Москва, Российская Федерация
ORCID: <https://orcid.org/0000-0001-9950-4716>
e-mail: yu_kan@mail.ru

Для цитаты:

Ардабьевский П.А., Гончар Д.А., Кан Ю.С. Статистическое моделирование ядра вероятностного распределения и его применение к решению задачи квантильной оптимизации с билинейной функцией потерь // Моделирование и анализ данных. 2020. Том 10. № 3. С. 69–84. DOI: <https://doi.org/10.17759/mda.2020100306>

**Ардабьевский Павел Александрович*, магистрант, Московский авиационный институт (национальный исследовательский университет), г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0002-6857-7485>, e-mail: ardpavel@gmail.com

***Гончар Дмитрий Александрович*, магистрант, Московский авиационный институт (национальный исследовательский университет), г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0002-2414-491X>, e-mail: dima.gonchar.29.08.13@gmail.com

****Кан Юрий Сергеевич*, д.ф.-м.н., профессор, Московский авиационный институт (национальный исследовательский университет), г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0001-9950-4716>, e-mail: yu_kan@mail.ru



В статье рассматривается плоская задача квантильной оптимизации с билинейной функцией потерь, которая с использованием достаточных условий оптимальности сводится к задаче линейного программирования. Сведение основано на использовании полиэдральной модели ядра вероятностного распределения вектора случайных параметров. Для построения этой модели предлагается алгоритм, основанный на методе статистического моделирования. Приводится описание программного комплекса построения модели ядра для ряда вероятностных распределений случайных параметров.

Ключевые слова: квантильная оптимизация, функция потерь, стохастическое программирование, ядро вероятностного распределения, минимаксная задача.

ВВЕДЕНИЕ

В настоящее время задачи квантильной оптимизации относятся к интенсивно развивающейся теории стохастического программирования с вероятностными критериями [1]. Функция квантили определяется как квантиль заданного уровня для вероятностного распределения некоторой функции потерь, зависящей от вектора оптимизируемых переменных и вектора случайных параметров. Оптимизационные модели с квантильным критерием используются в приложениях для учета риска или требований надежности при наличии неопределённости стохастической природы. В теории стохастического программирования задачи с билинейными функциями потерь занимают особое место, поскольку порождены экономической проблематикой – задачами оптимизации портфельных инвестиций с учетом риска [2–4].

Явные аналитические выражения для функции вероятности в содержательных практических задачах, как правило, получить не удастся. Более того, часто не удастся даже вычислить значения вероятностных критериев с любой наперед заданной точностью. В силу указанных причин при разработке численных методов оптимизации для этих критериев приходится использовать различные их аппроксимации. Например, аппроксимация может строиться на основе построения детерминированных границ для рассматриваемых функций. Суть предлагаемого в статье метода в том, что при выполнении некоторых достаточных условий оптимальности [5] задача квантильной оптимизации оказывается эквивалентной минимаксной, где максимум берется от функции потерь на ядре вероятностного распределения вектора случайных параметров, а минимум – по оптимизируемой стратегии. Ядро определяется как пересечение всех выпуклых и замкнутых доверительных множеств в пространстве реализаций этого вектора. Этим используемый в статье подход отличается от подхода работы [4], где вместо достаточных условий оптимальности предлагаются более жесткие и трудно проверяемые требования регулярности ядра. В статье предлагается способ построения сколь угодно точных статистических аппроксимаций ядра с помощью выпуклых многогранников. В этом заключается отличие предлагаемого способа аппроксимации ядра от алгоритмов, реализованных в [6], где предлагаются детерминированные полиэдральные аппроксимации. Использование этих статистических ап-



проксимаций вместо ядра в указанной выше минимаксной задаче, как и в [4], приводит к задаче линейного программирования с большим числом ограничений.

Целью настоящего исследования является отработка альтернативных алгоритмов моделирования ядра с целью развития результатов работы [4], а также разработка программного комплекса, реализующего эти алгоритмы, позволяющего исследовать их свойства на различных примерах академического характера.

ПОСТАНОВКА ЗАДАЧИ

Введем ряд обозначений. Пусть η – случайная величина с функцией распределения $F(x) = \tilde{N}(\eta \leq x)$. Квантилью заданного уровня $\alpha \in (0, 1)$ называется величина $[\eta]_\alpha = \min\{x : F(x) \geq \alpha\}$.

Рассматривается задача квантильной оптимизации:

$$\varphi_\alpha(u) = [u_0 b + u_1 \xi_1 + u_2 \xi_2]_\alpha \rightarrow \min_{\substack{u_0 \geq 0, u_1 \geq 0, u_2 \geq 0 \\ u_0 + u_1 + u_2 = 1}}$$

где $u = (u_0, u_1, u_2)$ – вектор оптимизируемых переменных, называемый ниже стратегией, $\xi = (\xi_1, \xi_2)$ – двумерный случайный вектор, компоненты которого независимы и одинаково распределены в соответствии с заданным законом распределения, $b \geq 0$ – детерминированная скалярная величина.

Данная задача может быть проинтерпретирована как экономическая задача формирования портфеля ценных бумаг с учетом риска. В этих терминах стратегия u определяет структуру инвестиционного портфеля, компоненты вектора ξ – доходности двух рисковых ценных бумаг, b – безрисковую ставку процента. При моделировании риска с помощью квантильного критерия считается [2,3], что он ограничивается допустимой величиной $1 - \alpha$.

ОСНОВНЫЕ ТЕОРЕТИЧЕСКИЕ РЕЗУЛЬТАТЫ

Алгоритмы решения поставленной задачи основаны на использовании понятия α -ядра распределения случайного вектора. Для случайного вектора ξ это ядро определяется соотношением

$$K_\alpha = \bigcap_{\|c\|=1} \left\{ x : c^T x \leq \left[c^T \xi \right]_\alpha \right\}.$$

Введем в рассмотрение функцию максимума

$$\psi_\alpha(u) = \max_{x \in K_\alpha} f(u, x),$$

где $f(u, x) = u_0 b + u_1 x_1 + u_2 x_2$, и вспомогательную минимаксную задачу

$$\psi_\alpha^* = \psi_\alpha(u_\alpha) = \min_{u \in U} \psi_\alpha(u),$$

в которой множество U допустимых стратегий то же самое, что и в рассматриваемой задаче квантильной оптимизации. В [1] установлено, что если ядро удовлет-



воряет некоторому условию регулярности, то $\varphi_\alpha(u) = \psi_\alpha(u)$ для всех $u \in U$. Поэтому вспомогательная задача оказывается эквивалентной исходной задаче квантильной оптимизации. Практическая проверка выполнения условия регулярности в настоящее время представляется нерешенной проблемой. Исключением является случай эллиптически симметричных распределений, указанный в [1]. В [5] доказано, что достаточным условием оптимальности решения вспомогательной задачи в исходной задаче квантильной оптимизации является выполнение условия

$$\tilde{N}(f(u_\alpha, \xi) \leq \psi_\alpha^*) \geq \alpha,$$

которое можно проверять, например методом Монте-Карло после решения вспомогательной задачи. При этом условие регулярности может быть не выполненным. Разработанный в рамках настоящего исследования и описанный ниже программный комплекс имеет своей целью провести апробацию такого подхода для различных законов распределения компонент случайного вектора ξ .

Приведем вспомогательную задачу к задаче линейного программирования (ЗЛП). Так как вектор u и константа b являются детерминированными, то

$$[u_0 b + u_1 \xi_1 + u_2 \xi_2]_\alpha = u_0 b + [u_1 \xi_1 + u_2 \xi_2]_\alpha.$$

Допустим, что α -ядро удалось аппроксимировать выпуклым полиэдром K_α^N . Впервые алгоритмы построения таких аппроксимаций предложены в [4] и реализованы в двумерном случае для ряда распределений в [6]. При замене -ядра его аппроксимацией во вспомогательной задаче получаем следующую минимаксную задачу

$$u_0 b + \max_{x \in K_\alpha^N} (u_1 x_1 + u_2 x_2) \rightarrow \min_{\substack{u_0 \geq 0, u_1 \geq 0, u_2 \geq 0 \\ u_0 + u_1 + u_2 = 1}}$$

которая согласно [4] эквивалентна ЗЛП вида

$$y \rightarrow \min_{\substack{u_0 \geq 0, u_1 \geq 0, u_2 \geq 0 \\ u_0 + u_1 + u_2 = 1 \\ u_0 b + u_1 x_{j1} + u_2 x_{j2} \leq y, j = \overline{1, N}}}$$

в которой $x_j = (x_{j1}, x_{j2})$, $j = \overline{1, N}$ – все вершины полиэдра K_α^N .

СТАТИСТИЧЕСКИЙ АЛГОРИТМ АППРОКСИМАЦИИ ЯДРА

Будем строить статистическую оценку полиэдральной аппроксимации -ядра в виде

$$K_\alpha^N = \bigcap_{i=1}^N \{x : c_i^T x \leq \widehat{[c_i^T \xi]_\alpha}\},$$

где «крышка» обозначает выборочную оценку квантили [1]. Для ее построения необходимо сначала сгенерировать векторы единичной нормали $c_i, i = \overline{1, N}$, где N – число ограничений. В [1] было предложено генерировать векторы c_i с помощью нанесения равномерной сетки на поверхность единичной окружности. Векторы c_i равны $(\cos \varphi_i, \sin \varphi_i)^T$, где φ_i – угол -той части разбиения окружности.



В отличие от работ [4,6] предлагается использовать случайную величину φ , равномерно распределенную по закону $R(0, 2\pi)$. Сгенерировав выборку угла φ объемом N , построим вариационный ряд полученных реализаций, а затем вычислим $\cos\varphi_i$ и $\sin\varphi_i$, где φ_i – i -ый член построенного вариационного ряда полученной выборки реализаций случайной величины φ .

Получив набор векторов единичной нормали c_i , построим выборочные оценки правых частей ограничений. Для этого генерируем выборку вектора ξ объемом L и для каждого вектора c_i построим вариационный ряд выборки $(c_i, \xi_j), j = 1, L$. Выборочной оценкой для каждого линейного ограничения будет служить αL -ый (функция округления дробного числа до ближайшего целого вправо) член соответствующего вариационного ряда. Таким образом получена система линейных ограничений.

Этот способ основывается на том, что при генерации векторов c_i был построен вариационный ряд случайной величины φ , потому что в этом случае мы получаем систему прямых, где каждая следующая прямая является «соседом» предыдущей, то есть если эти прямые являются сторонами аппроксимирующего многогранника, то их точка пересечения будет вершиной этого многогранника или, если по-другому, они являются соседними сторонами. К сожалению, из-за того, что правые части линейных ограничений вычисляются не точно, а в соответствии с выборочными (статистическими) оценками, то некоторые ограничения могут оказаться лишними. В результате число вершин аппроксимирующего полиэдра может оказаться меньше N , что отличает предлагаемый алгоритм от алгоритмов [4,6] и существенно затрудняет исследование его сходимости.

Итак, имеется множество линейных ограничений $c_i^T x \leq \widehat{[c_i^T \xi]}_\alpha, i = \overline{1, N}$, полученных в результате алгоритма генерации линейных ограничений. Прямые $c_i^T x = \widehat{[c_i^T \xi]}_\alpha, i = \overline{1, N}$, которые задают эти линейные ограничения, являются сторонами аппроксимирующего многогранника, причем они расположены в таком порядке, что каждая следующая прямая $c_{i+1}^T x = \widehat{[c_{i+1}^T \xi]}_\alpha$ является «соседом» предыдущей. Это обстоятельство позволяет серьезно снизить количество решаемых систем линейных уравнений по сравнению с алгоритмом, используемым в [4].

Шаг 1. Составляем и решаем системы линейных уравнений $c_i^T x = \widehat{[c_i^T \xi]}_\alpha$ (для первой итерации алгоритма $i = 1$) и прямых $c_j^T x = \widehat{[c_j^T \xi]}_\alpha, j = i + 1, N$. Решаем до тех пор, пока не будет найдена точка пересечения, удовлетворяющая всем линейным ограничениям, или пока не переберем все возможные системы. Если подходящая точка была найдена, то прямые $c_k^T x$ и $c_k^T x = \widehat{[c_k^T \xi]}_\alpha$, где k – номер первой встретившейся прямой, точка пересечения с которой удовлетворяет всем линейным ограничениям, являются сторонами многогранника, их точка пересечения является его вершиной, а все прямые $c_i^T x = \widehat{[c_i^T \xi]}_\alpha, i = \overline{2, k-1}$, не являются сторонами аппроксимирующего многогранника, переходим к шагу 1, $i = k$. Если подходящих точек найдено не было, и прямая ранее не была отмечена как сторона аппроксимирующего многогранника, переходим к шагу 2. Если же для нее уже есть одна подходящая точка, переходим к шагу 3.



Шаг 2. Если прямая $c_i^T x = \left[c_i^T \xi \right]_\alpha$ не была ранее отмечена как сторона многогранника, то эта прямая ей не является. Переходим к шагу 1, $i = i + 1$.

Шаг 3. Если ранее для прямой $c_i^T x = \left[c_i^T \xi \right]_\alpha$ была найдена одна вершина, то есть эта прямая является стороной аппроксимирующего многогранника, пересекаем эту прямую с самой первой подошедшей прямой. Найденная точка будет являться вершиной искомого многогранника. Конец алгоритма.

Как видно в этом алгоритме, мы не решаем все C_n^2 линейных систем уравнений, как это было в [4] даже в самом плохом случае. При высоком объеме выборки L случайного вектора ξ ниже на примерах увидим, что сторонами аппроксимирующего многогранника являются практически все или все прямые, задающие линейные ограничения. В таком случае количество решаемых линейных систем становится равно N .

ОПИСАНИЕ ПРОГРАММНОГО КОМПЛЕКСА

Рассмотрим интерфейсное окно:

Количество векторов

Количество ограничений

Уровень надежности на отрезке [0; 1]

Детерминированная константа

Логнормальное распределение
 Нормальное распределение
 Равномерное распределение
 Кусочное равномерное
 Распределение Коши
 Экспоненциальное распределение
 Гамма распределение
 Распределение Рэлея
 Распределение Вейбулла

Первый параметр Второй параметр

Для расчета объема выборки

Бетта Эпсилон

Пуск

В интерфейсе требуется указать следующее:

- Количество векторов – количество сгенерированных случайным образом векторов, из которых будет строиться ядро
- Количество ограничений – количество случайным образом сгенерированных векторов распределения $R(0, 2\pi)$
- Уровень надежности – квантиль полученной выборки векторов
- Детерминированная константа – постоянная величина, используемая при решении ЗЛП
- Выбор одного из распределений
- Ввод параметров распределения
- Бетта и эпсилон – величины, используемые для расчёта рекомендуемого объема выборки
- Пуск – кнопка запуска расчёта

Вывод программы, а также описание выводимых полей будет приведено в «4. Результаты работы программы»

Проанализируем время работы программы.

Так как пользователь будет желать получить точную аппроксимацию, то он будет вводить большое количество векторов и ограничений. Рассмотрим таблицу времени генерации векторов и ограничений при разных задаваемых параметрах:

Рис. 3.1. Окно ввода данных для работы программы



Таблица 3.1

Генерация векторов и ограничений

Кол-во векторов	Кол-во ограничений	Время генерации (сек)
10 ⁴	100	0.28
10 ⁴	200	0.46
10 ⁴	300	0.6
10 ⁵	100	39
10 ⁵	200	67.57
10 ⁵	300	91.17
10 ⁶	100	496.17
10 ⁶	200	890.92
10 ⁶	300	1311.97

Опираясь на результаты, приведенные в таблице 3.1, можно сказать, что оба вводимых параметра значимо влияют на время работы. Далее мы скажем оптимальные параметры для получения результата. Сейчас скажем только то, что кол-во векторов должно варьироваться от 10⁴ до 10⁶, так как при меньших значениях отбрасывается слишком много прямых, а при больших время работы программы слишком велико.

Теперь проведем анализ сложности отбрасывания прямых. По данному шагу были проанализированы два варианта отброса.

В первом варианте нам потребовалось перебрать все точки пересечения прямых (сложность $O(\frac{N!(N-1)!}{2})$), для каждой точки необходимо было решить систему (сложность $O(N^2)$), а затем отсортировать полученные вершины (сложность $O(N)$).

Во втором случае мы лишь сортируем прямые (сложность $O(N)$) и находим точки проходя по всем прямым (сложность $O(N)$).

Таблица ниже демонстрирует разницу в работе:

Таблица 3.2

Нахождение вершин многогранника

Кол-во векторов	Кол-во ограничений	Время нахождения вершин (без сортировки прямых / с сортировкой прямых) (сек)
10 ⁴	100	7.34 / 2.58
10 ⁴	200	41.04 / 16.76
10 ⁴	300	234.19 / 53.45
10 ⁵	100	6.32 / 2.04
10 ⁵	200	41.81 / 17.08
10 ⁵	300	240.75 / 52.14
10 ⁶	100	7.2 / 2.78
10 ⁶	200	42.89 / 16.95
10 ⁶	300	54.96 / 18.83

Из таблицы можно заметить, что количество введенных векторов не влияет на время отбрасывания прямых.



Для краткости описания введем обозначения:

- L – количество векторов
- N – количество ограничений N

Рассмотрим теперь таблицу, в которой приведены итоги работы конечной программы:

Таблица 3.3

Результат работы

L	N	Время генерации (сек)	Время отброса прямых и нахождения вершин (сек)	Кол-во вершин многогранника
10^4	100	0.28	2.58	63
10^4	200	0.46	16.76	72
10^4	300	0.6	53.45	102
10^5	100	39	2.04	76
10^5	200	67.57	17.08	108
10^5	300	91.17	52.14	127
10^6	100	496.17	2.78	88
10^6	200	890.92	16.95	149
10^6	300	1311.97	54.96	189

Также нельзя не заметить, что время построения аппроксимации ядра всегда варьируется в пределах [5, 6] секунд

Оптимальные параметры пользователь может выбрать таким образом: если более важно время работы программы, то стоит выбрать $L=10^5$ и $N=250$; если более важна точность, то выбрать $L=10^6$ и $N=200$.

РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММЫ

В результате работы мы можем увидеть следующее: (см. рис. 4.1)

Опишем выводимые колонки.

Первая колонка – повторный ввод данных. Она необходима для изменения введенных параметров.

Вторая колонка – результаты решения ЗЛП с дополнительными данными. Опишем их:

Таблица – значения u_0 , u_1 , u_2

Значение целевой функции – значение при расчёте ЗЛП

Расчетное значение вероятности по методу Монте-Карло

Строка, в которой написано является ли решение оптимальным

Количество векторов – введенное количество векторов

Количество вершин многогранника – количество вершин ядра, получившееся в результате отброса прямых ограничениями

Следующие три строки описывают время работы программы

N , Nr – оптимальные размеры выборки

Третья колонка отображает построенное ядро

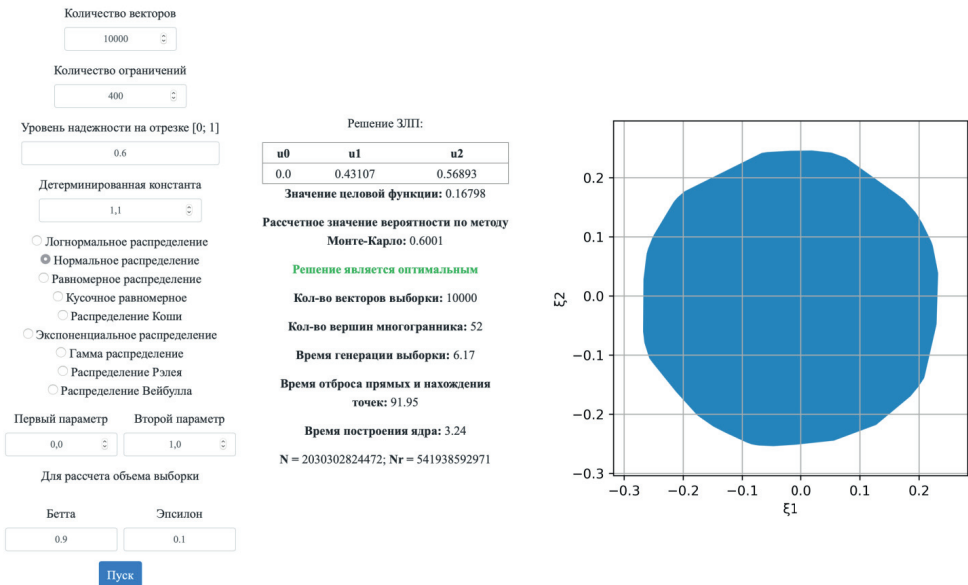


Рис. 4.1. Результат работы программного комплекса

Рассмотрим случай равномерного распределения компонент вектора ξ на отрезке $(-1, 1)$, $\alpha = 0.9$, количество ограничений $N = 50$.

На рис. 4.2 и рис. 4.3 видно, что с увеличением объема выборки L увеличивается и точность аппроксимации хоть и несильно, и результат сходится к аналитическому результату. Если бы в разработанной программе была возможность настраивать распределение каждой компоненты вектора ξ отдельно, то схожесть форм в случае равномерного распределения компонент аналитически построенного аппроксимирующего многогранника и построенного на основе выборочной оценки была бы заметна куда больше.

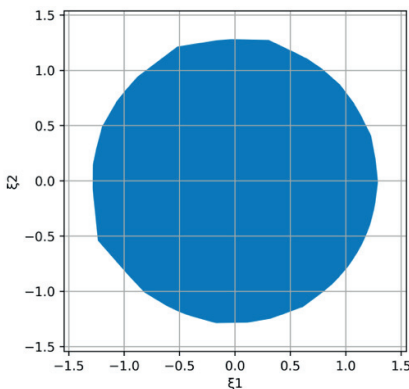


Рис. 4.2 Результат аппроксимации для равномерного распределения, $L = 10^4$

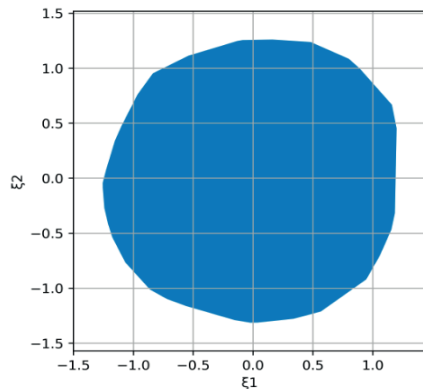


Рис. 4.3 Результат аппроксимации для равномерного распределения, $L = 10^6$



Теперь построим аппроксимации для нормального распределения $N(0,1)$ компонент вектора ξ , $\alpha = 0.9$, количество ограничений $N = 50$.

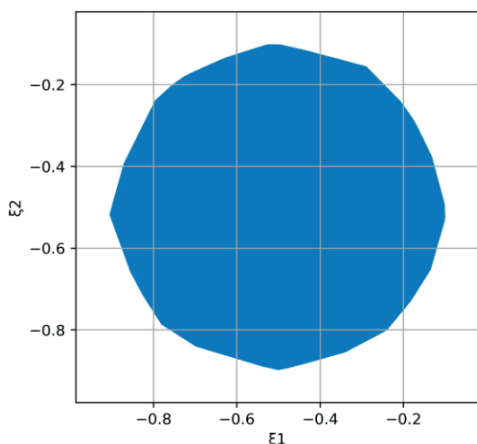


Рис. 4.4 Результат аппроксимации для нормального распределения, $L = 10^4$

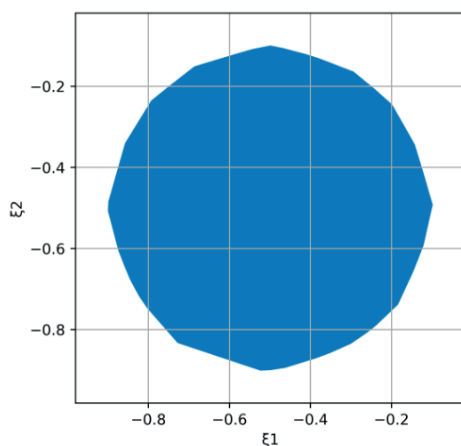


Рис. 4.5 Результат аппроксимации для нормального распределения, $L = 10^6$

Как видно на рис. 4.4 и рис. 4.5 результат так же, как и в случае с равномерным распределением, все больше становится похож на полученный аналитически с увеличением объема выборки L . Даже неровность формы из-за случайной природы векторов c_i сглаживается и сходится к ровному кругу.

Результат для распределения Коши отличается по своему характеру от результатов, полученных для равномерного и нормального распределений. Было использовано стандартное распределение Коши для обеих компонент вектора ξ , $\alpha = 0.9$, $N = 100$.

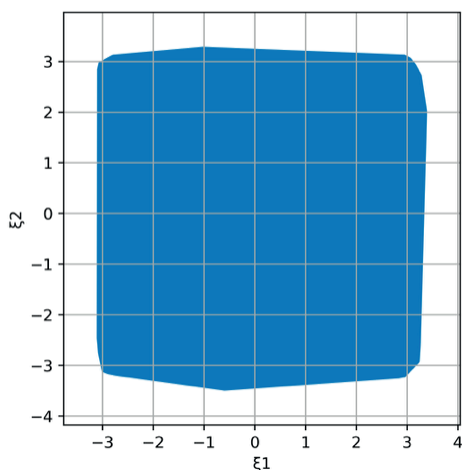


Рис. 4.6 Результат аппроксимации для распределения Коши, $N = 50$, $L = 10^4$

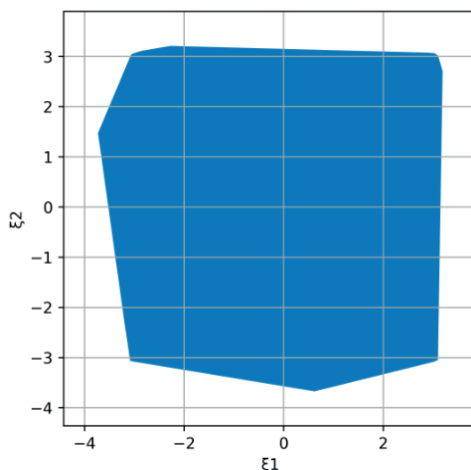


Рис. 4.7 Результат аппроксимации для распределения Коши, $N = 50$, $L = 10^5$

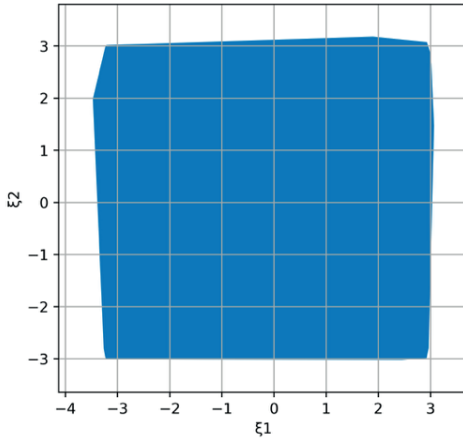


Рис. 4.8 Результат аппроксимации для распределения Коши, $N=100$, $L=10^4$

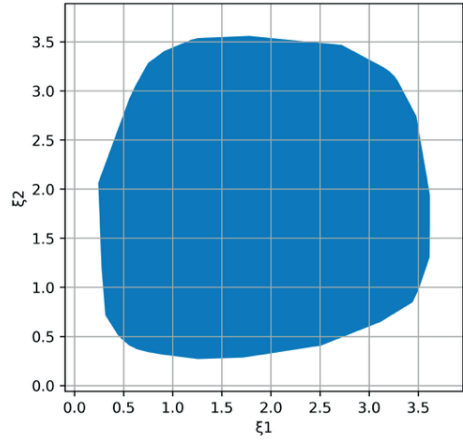


Рис. 4.9 Результат аппроксимации для логнормального распределения, $L=10^4$

Как оказалось, аппроксимация в случае распределения Коши плохо реагирует на повышение объема выборки L , этот эффект мы можем наблюдать, сравнив рис. 4.6 и рис. 4.7. Зато увеличение количества ограничений N увеличивает точность аппроксимации, и результат сходится к полученному аналитическому.

Аппроксимация при логнормальном и экспоненциальном распределении компонент вектора ξ схожа по своему характеру с аппроксимацией для равномерного и нормального распределений. Для построения многогранников были использованы $\text{Log}N(1)$ и $E(1)$, $\alpha = 0.9$, количество ограничений $N = 50$.

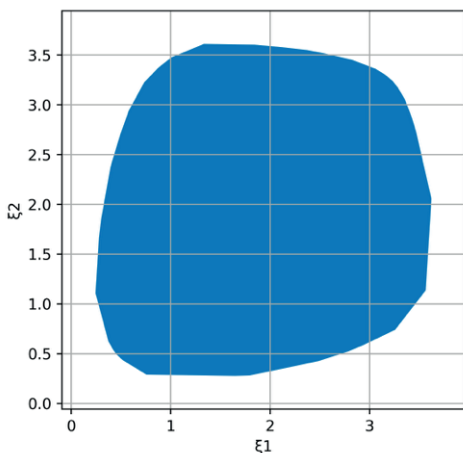


Рис. 4.10 Результат аппроксимации для логнормального распределения, $L=10^6$

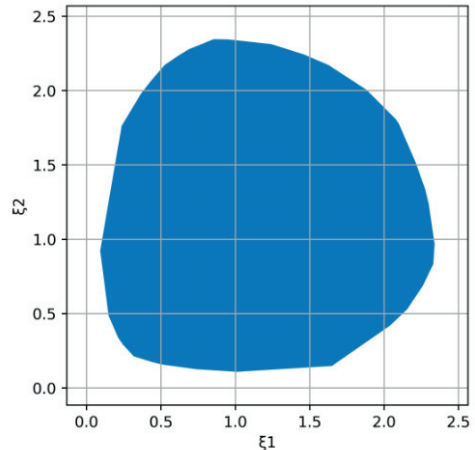


Рис. 4.11 Результат аппроксимации для экспоненциального распределения, $L=10^4$

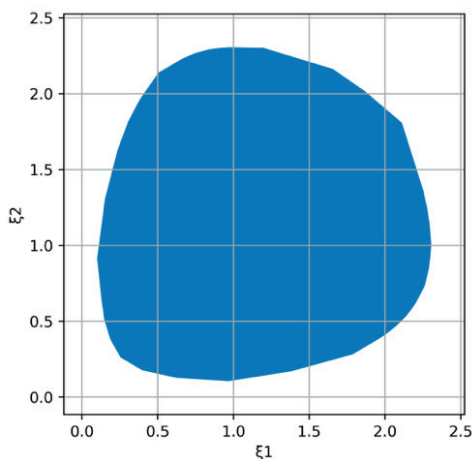


Рис. 4.12 Результат аппроксимации для экспоненциального распределения, $L=10^6$

Из полученных результатов видна правильность работы алгоритма, а также эмпирическим способом была доказана сходимость густой сетки векторов единичной нормали к истинным α -ядрам. Стоит отметить, что α не варьировалась по причине того, что она влияет только на размер α -ядра, а не на его форму (меняется только для равномерного распределения).

Ниже приведены результаты решения исходной задачи для $b=1.1$, $N=50$, (объем выборки был взят небольшим, чтобы уменьшить время работы программы), компоненты вектора ξ распределены равномерно на $(-1,1)$. Ответом на поставленную задачу является вектор стратегий u .

Решение ЗЛП:

u_0	u_1	u_2
0.08663	0.91337	0.0

Значение целевой функции y : 0.0

Кол-во векторов выборки: 10000

Кол-во вершин многогранника: 41

Время генерации выборки: 2.91

Время отброса прямых и нахождения точек: 0.38

Время построения ядра: 5.34

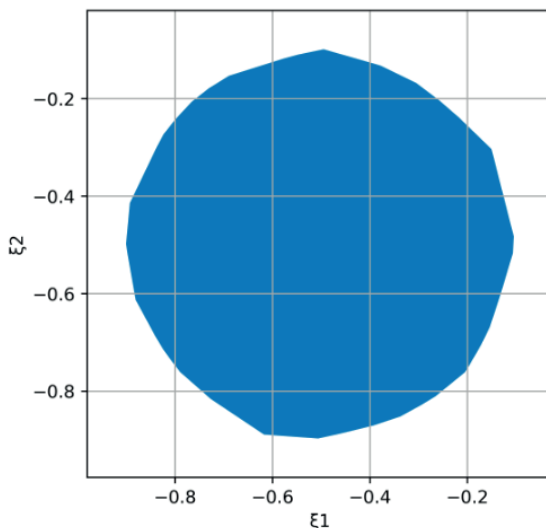


Рис. 4.13 Решение задачи при $\alpha = 0.9$



Решение ЗЛП:

u0	u1	u2
0.0	0.52357	0.47643

Значение целевой функции y : 0.76047

Кол-во векторов выборки: 10000

Кол-во вершин многогранника: 40

Время генерации выборки: 2.6

Время отброса прямых и нахождения точек: 0.32

Время построения ядра: 5.04

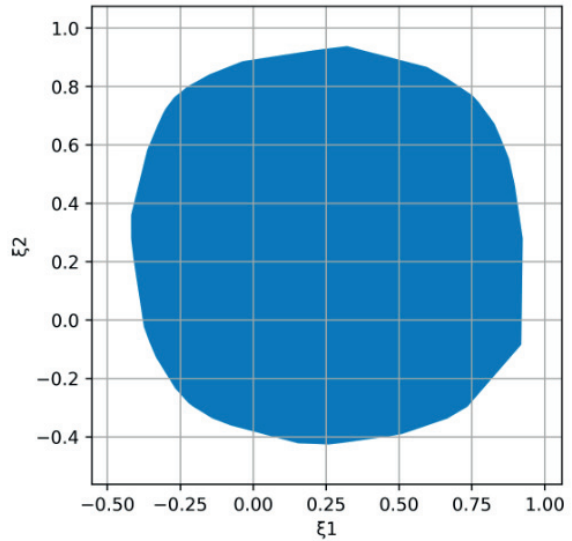


Рис. 4.14 Решение задачи при $\alpha = 0.95$

Решение ЗЛП:

u0	u1	u2
0.00178	0.99822	0.0

Значение целевой функции y : 0.0

Кол-во векторов выборки: 10000

Кол-во вершин многогранника: 32

Время генерации выборки: 2.92

Время отброса прямых и нахождения точек: 0.37

Время построения ядра: 5.23

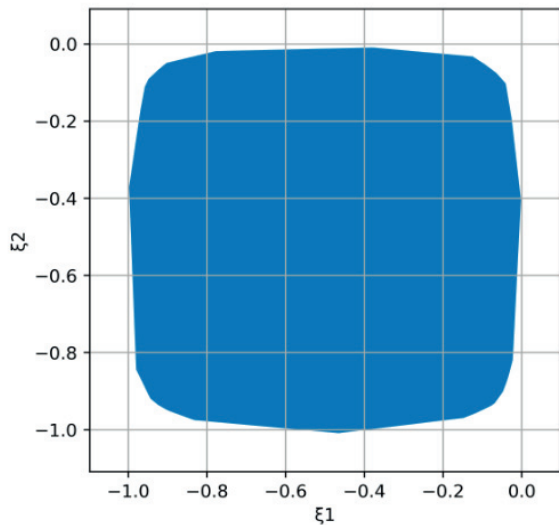


Рис. 4.15 Решение задачи при $\alpha = 0.99$

На вышеприведенных рис. 4.13, рис. 4.14 и рис. 4.15 отчетливо видно, как с ростом α форма α -ядра становится все более квадратной. Это связано с тем, что случайный



вектор ξ распределен равномерно на квадрате, и так как α -ядро с ростом α увеличивается, а квадрат, на котором распределен вектор ξ , нет, получается такая форма.

ЗАКЛЮЧЕНИЕ

В результате проделанной работы разработана программа для решения задачи квантильной оптимизации с билинейной функцией потерь с заданным распределением случайных параметров при помощи построения аппроксимации ядра различной точности.

В программе имеются следующие изменяемые пользователем параметры:

- Количество векторов;
- Количество ограничений;
- Уровень надежности;
- Значение детерминированной константы;
- Распределение;
- Параметры распределения;
- Параметры расчёта оптимального числа векторов и ограничений.

Результаты многочисленных расчетов для различных распределений свидетельствуют о конкурентоспособности используемого статистического алгоритма аппроксимации -ядра по сравнению с алгоритмами [4,6].

Литература

1. Кибзун А.И., Кан Ю.С. Задачи стохастического программирования с вероятностными критериями. – М.: Физматлит, 2009.
2. Кан Ю.С. Формирование портфелей ценных бумаг с учетом риска. Учеб. пособие. – М.: Изд-во МАИ, 2008.
3. Кан Ю.С., Тузов Н.В. Минимизация квантили нормального распределения билинейной функции потерь. // Автоматика и телемеханика. 1998, № 11, с. 82–92.
4. Васильева С.Н., Кан Ю.С. Метод решения задачи квантильной оптимизации с билинейной функцией потерь // Автоматика и телемеханика, 2015, № 9, с. 83–101.
5. Кан Ю.С. Расширение задачи квантильной оптимизации с линейной по случайным параметрам функцией потерь // Автоматика и телемеханика, 2020, № 12 (в печати).
6. Васильева С.Н., Кан Ю.С. Алгоритм визуализации плоского ядра вероятностной меры. // Информатика и ее применения, 2018, № 2, с. 60–68.



Statistical Simulation of the Probability Distribution Kernel and Its Application to Solve the Quantile Optimization Problem with the Bilinear Loss Function

Pavel A. Ardabyevskiy*

Moscow Aviation Institute (National Research University), Moscow, Russia
ORCID: <https://orcid.org/0000-0002-6857-7485>
e-mail: ardpavel@gmail.com

Dmitry A. Gonchar**

Moscow Aviation Institute (National Research University), Moscow, Russia
ORCID: <https://orcid.org/0000-0002-2414-491X>
e-mail: dima.gonchar.29.08.13@gmail.com

Yuri S. Kan***

Moscow Aviation Institute (National Research University), Moscow, Russia
ORCID: <https://orcid.org/0000-0001-9950-4716>
e-mail: yu_kan@mail.ru

The article considers a plane quantile optimization problem with a bilinear loss function, which, using sufficient optimality conditions, is reduced to a linear programming problem. The reduction is based on the use of a polyhedral model of the kernel of the probability distribution of the vector of random parameters. To build this model, an algorithm based on the method of statistical modeling is proposed. A description of the software package for constructing a kernel model for a number of probability distributions of random parameters is given.

Keywords: quantile optimization, loss function, stochastic programming, kernel of probability distribution, minimax problem.

For citation:

Ardabyevskiy P.A., Gonchar D.A., Kan Y.S. Statistical Simulation of the Probability Distribution Kernel and Its Application to Solve the Quantile Optimization Problem with the Bilinear Loss Function. *Modelirovanie i analiz dannykh = Modelling and Data Analysis*, 2020. Vol. 10, no. 3, pp. 69–84. DOI: <https://doi.org/10.17759/mda.2020100306> (In Russ., abstr. in Engl.).

***Pavel A. Ardabyevskiy**, Student of the magistracy, Moscow Aviation Institute (National Research University), Moscow, Russia, ORCID: <https://orcid.org/0000-0002-6857-7485>, e-mail: ardpavel@gmail.com

****Dmitry A. Gonchar**, Student of the magistracy, Moscow Aviation Institute (National Research University), Moscow, Russia, ORCID: <https://orcid.org/0000-0002-2414-491X>, e-mail: dima.gonchar.29.08.13@gmail.com

*****Yuri S. Kan**, Dr.Sci.(Phys.-Math.), professor, Moscow Aviation Institute (National Research University), Moscow, Russia, ORCID: <https://orcid.org/0000-0001-9950-4716>, e-mail: yu_kan@mail.ru



References

1. Kibzun A.I., Kan Y.S. Zadachi stohasticheskogo programmirovaniya s veroyatnostnymi kriteriyami. – M.: Fizmatlit, 2009.
2. Kan Y.S. Formirovanie portfelej cennyh bumag s uchetom riska. M.: MAI publishing, 2008.
3. Kan Y.S., Tuzov N.V. Minimizatsiya kvantili normal'nogo raspredeleniya bilinejnoj funkicii poter'. Avtomatika i Telemekhanika. 1998, № 11, pp. 82–92.
4. Vasil'eva S.N., Kan Y.S. Metod resheniya zadachi kvantil'noj optimizatsii s bilinejnoj funkciej poter'. Avtomatika i Telemekhanika, 2015, № 9, pp/ 83–101.
5. Kan Y.S. Rasshirenie zadachi kvantil'noj optimizatsii s linejnoj po sluchajnym parametram funkciej poter'. Avtomatika i Telemekhanika, 2020, № 12.
6. Vasil'eva S.N., Kan Y.S. Algoritm vizualizatsii ploskogo yadra veroyatnostnoj mery. Informatika i ee primeneniya, 2018, № 2, pp. 60–68.

◇◇◇◇◇◇◇◇◇◇ **МЕТОДИКА ОБУЧЕНИЯ** ◇◇◇◇◇◇◇◇◇◇

УДК 377.131.11; 377.131.14; 377.169.3

**Особенности программной реализации
цифровой адаптивной платформы для обучения
программированию с функцией формирования
навыков командной работы**

Попков С.И.*

Московский государственный психолого-педагогический
университет, г. Москва, Российская Федерация
ORCID: <https://orcid.org/0000-0003-2566-1262>
e-mail: rslw25@gmail.com

Проведено исследование имеющихся подходов к построению образовательных сред для обучения программированию. Проанализированы их основные недостатки. Предложена концепция цифровой адаптивной платформы для обучения программированию, учитывающая выявленные недостатки. Описаны особенности программной реализации в виде общей архитектуры, принципов работы и построения взаимосвязей компонентов платформы. Продемонстрированы особенности программной архитектуры и реализации цифровой адаптивной платформы для обучения программированию с функцией формирования навыков командной работы.

Ключевые слова: адаптивность, цифровая платформа, обучение программированию, навыки командной работы.

Для цитаты:

Попков С.И. Особенности программной реализации цифровой адаптивной платформы для обучения программированию с функцией формирования навыков командной работы // Моделирование и анализ данных. 2020. Том 10. № 3. С. 85–100. DOI: <https://doi.org/10.17759/mda.2020100307>

***Попков Сергей Игоревич**, кандидат физико-математических наук, доцент факультета информационных технологий, заведующий лабораторией, Московский государственный психолого-педагогический университет, г. Москва, Российская Федерация, ORCID: <https://orcid.org/0000-0003-2566-1262>, e-mail: rslw25@gmail.com



ВВЕДЕНИЕ

Вопрос разработки и внедрения специализированных программных средств для обучения программированию актуален с конца XX века, с момента внедрения программирования в частности и информатики в целом как дисциплины, преподаваемой в учебных заведениях разного уровня – в школах, колледжах, высших и других учебных заведениях. Такие средства ставят своей задачей реализовать надежную с точки зрения информационной безопасности среду, при использовании которой обучающийся использует язык программирования (как правило, с ограниченным подмножеством доступных операторов и функций из некоторого реального языка) для воздействия на вычислительную среду и достижения заданного результата, тем самым закрепляя необходимые знания о процессе программирования и практике его применения на задачах, приближенных к реальным.

Существуют два наиболее часто используемых подхода к разработке подобных обучающих средств. Один из них основан на создании ряда предъявляемых заданий, оформленных в виде уровней с линейно растущей сложностью. Как правило, действия на каждом из таких уровней заранее ограничены некоторым набором возможных операций, чтобы упростить задачу обеспечения информационной безопасности и надежности обучающей среды. Примером такой среды является игра «Кодвардс» [1].

Второй подход подразумевает генерируемый свободный мир (также именуемый «песочницей», англ. “sandbox”), в котором обучающийся постепенно расширяет круг своих возможностей взаимодействия с миром за счет решения все более сложных задач. На финальном этапе обучающемуся доступен максимально широкий спектр возможностей для решения самой сложной задачи либо конечная цель задается через комплексную совокупность решения ряда более простых задач. Таким образом, мир на программном уровне обеспечивает полный набор возможных операций над ним с самого начала взаимодействия и взаимосвязанные задачи разного уровня сложности, но они закрыты для участника образовательного процесса до тех пор, пока он не разблокирует доступ к ним через решение задач соответствующего уровня сложности. Примером такой платформы является игра «CoLoBot» [2].

И в том, и в другом случае построение обучающей среды с обеспечением всех мер информационной безопасности, разработка транслятора для предоставляемого пользователю подмножества языка программирования, создание и оцифровка сценариев и графических элементов представляет собой набор достаточно трудоемких задач с высоким уровнем сложности их разработки как по отдельности, так и в качестве взаимосвязанных компонентов в составе обучающего комплекса, поэтому зачастую в таких средствах обучения нет элемента командного взаимодействия, который вносит свой вес в каждую составляющую критерия сложности разработки. Тем не менее, последние прорывы в области глубинного обучения [3] позволяют актуализировать вопрос разработки такой среды, представляющей собой цифровую платформу для обучения программированию с функцией формирования навыков командной работы.



Кроме того, зачастую задания, представленные в подобных платформах, строятся по принципу линейного увеличения сложности без учета реальных способностей обучающегося: предполагается что, если пользователь платформы прошел более легкое задание, он готов к следующему поступательно предъявляемому сложному, что не всегда верно. Возникает необходимость в применении адаптивного подхода к предъявлению заданий. Механизм, реализующий такой подход, был продемонстрирован в тренажере для операторов сложных систем [4, 5], однако он был ориентирован на предъявление заданий различного уровня сложности без учета характера ошибок, допущенных оператором. В данной работе необходимо преодолеть это ограничение алгоритма, поскольку игнорирование характера ошибки не позволяет обучающемуся проработать соответствующие разделы дисциплины, связанные с ними навыки и компетенции, что ведет к некачественному уровню понимания фундаментальных понятий дисциплины и образовательного процесса в целом.

Цель данной работы – исследовать имеющиеся подходы к построению образовательных сред для обучения программированию, выявление их недостатков и предложение собственного решения, позволяющего нивелировать или сгладить выявленные недостатки.

Актуальность материала, представленного в данной статье, обусловлена не только востребованностью решения для поставленной задачи, но и предлагаемыми способами практической реализации дополнительных возможностей, востребованных в образовательной сфере, таких как адаптивность предъявляемых заданий и формирование навыков командной работы у обучающихся.

ОБЗОР ПОПУЛЯРНЫХ ПОДХОДОВ К РЕАЛИЗАЦИИ СРЕДЫ ДЛЯ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ

Применение цифровых обучающих технологий – один из актуальных подходов к ведению образовательного процесса, который принимает новые формы в связи с развитием программно-аппаратных средств и новейших стандартов веб-технологий. Появление современных систем управления курсами, как коммерческих, так и свободных (Moodle [6], edX [7], Chamilo [8] и др.), а также средств проведения тестирования (FlexiQuiz [9], ClassMarker [10], комплекс программных продуктов компании Pesofts [11]) позволяет реализовать гибкое окружение для ведения преподавательской деятельности и оценки уровня знаний студентов в различных сферах, таких, как математика или гуманитарные науки.

Однако перечисленные средства не учитывают специфику обучения программированию, которое представляет собой специфический процесс и требует определенных технологий, обеспечивающих оценку качества программного кода. Особенность такого рода программного обеспечения заключается в представлении взаимосвязанных заданий в игровой форме, призванной пробудить интерес к обучению, и форме ответа, представляющей собой свободный программный код, который выполняется в безопасной для вычислительной системы среде, автоматически проверяется на кор-



ректность и позволяет достичь поставленной в задаче цели множеством способов без необходимости выбора конкретного ответа или группы ответов (в отличие от классического подхода к тестированию).

Далее будут кратко рассмотрены наиболее известные и востребованные программные продукты, специализирующиеся на обучении программированию с учетом особенностей процесса обучения.

SberCraft

SberCraft – платформа, выпущенная группой компаний «Сбер» при поддержке компании «Gecko» [12]. Представляет собой игру в жанре RPG с линейным сюжетом, в которой на основе ряда параметров (языка программирования, уровня коммерческого опыта и т.д.). Цель – формирование компетенций, необходимых группе компаний «Сбер» для приема на работу, а также предварительная оценка квалификации кандидатов перед собеседованием. Может использоваться в качестве тренажера для программистов.

Среди сильных сторон можно отметить красивую реализацию, учитывающую целевую аудиторию, а также нетривиальные задачи с ограничением по времени их выполнения, богатая палитра языков (на момент написания статьи поддерживаются 12 языков программирования).

Индивидуальный недостаток: платформа заточена под рекрутинг и служит интересам конкретных компаний.

Недостатки, общие для всех программных продуктов, специализирующихся на обучении программированию, будут приведены ниже в разделе 2.

Кодвардс

Кодвардс – онлайн-платформа для обучения детей основам программирования в игровой форме. Разработана компанией «Redmadrobot» [13] с привлечением специалистов – детских педагогов, психологов, специалистов ИТ-индустрии. Представляет собой игру в двумерном пространстве, в которой агент (робот) решает задачи, выполняя команды упрощенного вымышленного языка программирования. Участник линейно переходит с уровня на уровень, поступательно увеличивая сложность и осваивая новые концепции программирования.

Среди сильных сторон можно отметить высокое качество графики, интуитивный интерфейс, подсказки, широкий охват материала, в том числе, основ объектно-ориентированной парадигмы программирования. Представлены задачи на поиск ошибок в коде.

Индивидуальный недостаток: платформа и ее содержимое ориентированы на учеников младшей школы.

CodinGame

Онлайн-платформа, разработанная одноименной компанией, «CodinGame» предлагает набор различных соревнований, нацеленных на совершенствование навыков программирования для всех уровней, включая опытных программистов [14]. Можно охарактеризовать как сборник стратегий в реальном времени с качественной трехмерной графикой, где для решения поставленных задач необходимо писать код для агента-исполнителя на реальном языке программирования.



Сильных сторон много, в их числе широкая палитра реальных языков программирования, множество предлагаемых тем, от простейших (например, циклы и условия) до сложных, ориентированных на опытных программистов (например, задачи на оптимизацию и многоагентные системы). Существует собственная система рекрутинга, позволяющая компаниям отбирать самых перспективных программистов [15]. Содержимое платформы постоянно пополняется новыми заданиями.

Индивидуальный недостаток: представляет собой набор разрозненных независимых заданий, не позволяющих учесть особенности учебной траектории лица, проходящего обучение.

CoLoBot

CoLoBot – (Colonize With Bots) – игра 2001 года, проходящая в данный момент процесс ремастеринга [16] (переиздания с учетом требований к современным программным продуктам). Это учебная стратегия в реальном времени для изучения программирования, включающая в себя миссии по колонизации различных планет с помощью роботов, управляемых программами на вымышленном языке программирования, близком по синтаксису к C++.

Сильные стороны проекта: свободный трехмерный мир, возможность создавать собственные карты (сценарии игры), интересные миссии (с единой сюжетной линией).

Индивидуальные недостатки: проект серьезно устарел, а предварительный релиз современного переиздания выполнен в феврале 2019 года (то есть на момент написания статьи прошло больше 1,5 лет с момента последнего предварительного издания, разработка идет достаточно медленно, последняя версия – 0.1.12). Продукт ориентирован на подростков, интересующихся программированием. Ограничения на объем входного кода не позволяют использовать серьезные возможности программирования, например, применение технологий искусственного интеллекта. Игра предусматривает режим, в котором может быть пройдена и без применения навыков программирования.

КЛЮЧЕВЫЕ НЕДОСТАТКИ РАССМОТРЕННЫХ ПОДХОДОВ

У рассмотренных проектов есть ряд существенных недостатков, которые важно устранить для более эффективного применения и развития образовательных платформ для обучения программированию.

Во-первых, ориентир на базу готовых заданий. Когда задания не генерируются и не модифицируются, становятся менее значимыми преимущества среды для написания произвольного кода, поскольку появляется возможность использовать чужой код у другого участника (например, студента старших курсов), то есть такой же «готовый ответ», как и в случае тестирования.

Во-вторых, линейность развития персонажа, с которым ассоциирует себя обучающийся. Сначала предъявляются наиболее простые задачи, затем – сложные. При этом необходимо пройти предъявляемую задачу текущего уровня сложности, чтобы выйти на следующий. Это, с одной стороны, приводит к снижению уровня вовлечен-



ности и интереса у более опытных участников на начальном этапе, а также создает риск «ловушки», когда участник не может пройти более сложное задание и продвигнуться вперед, что может привести к «выгоранию». Было бы полезно в таких случаях предусмотреть адаптивные упражнения, которые позволили бы преодолеть «разрыв» между уровнями: проанализировать неверные решения участника и отработать конкретные ситуации, вызывающие сложность (далее этот принцип для краткости будет называться «микроадаптивность»).

В-третьих, преимущественно индивидуальный подход в оценке уровня подготовки, отсутствие задач коллективного типа с ориентацией на развитие навыков работы в команде, что особенно важно для любого современного специалиста в IT-индустрии.

В-четвертых, сетевые проекты требуют больших затрат на безопасность, чтобы избежать «утечки» банка данных или нежелательного изменения хранилища данных с результатами работы участников. Для работы в аудитории есть возможность существенно упростить архитектуру без снижения уровня безопасности, что, в свою очередь, позволяет уделить время и другие ресурсы на разработку более полезных элементов платформы, например, функций формирования навыков командной работы.

В-пятых, в дистанционных платформах обучения отсутствует очное взаимодействие с преподавателем. Удобное переносимое архитектурное решение, допускающее локальное развертывание платформы, позволило бы не только отслеживать факты мошенничества и злоупотребления платформой со стороны студентов, но и, по усмотрению преподавателя, допускать его участие и советы по решению конкретного участка задачи. При этом, при необходимости, программная архитектура платформы допускает возможность перевода состояния среды в дистанционный режим.

В-шестых, статичные задания не только увеличивают вероятность списывания, но и не позволяют построить систему, которая адаптивно подбирала бы траекторию задач и с учетом уровня конкретного обучающегося индивида, и с учетом уровня группы в целом. Тогда как правильно обученная модель машинного обучения может более тонко подогнать динамически изменяемые параметры на основе статистических данных об успехах конкретного студента, сделав переход между уровнями навыков менее дискретным.

В-седьмых, платформы не всегда учитывают особенности студентов с ограниченными возможностями для равноправной оценки уровней всех групп студентов. К примеру, для защиты от копирования условие задачи может быть представлено в форме монолитного изображения, недоступного для распознавания программными средствами для лиц с нарушениями зрения. В подобной ситуации, при индивидуальном подходе, можно учесть особенности и представить такому лицу подходящую по форме задачу того же уровня. Это может быть особенно актуально для инклюзивных групп в режиме командной работы. Для этого должен быть реализован редактор, позволяющий вносить изменения в генерируемые блоки заданий и их предпросмотр.

Для того, чтобы устранить выявленные недостатки, необходимо создать более комплексную платформу с особыми характеристиками.

ПОСТАНОВКА ЗАДАЧИ И ФОРМУЛИРОВАНИЕ УСЛОВИЙ, КОТОРЫМ ДОЛЖНА СООТВЕТСТВОВАТЬ ЦИФРОВАЯ АДАПТИВНАЯ ПЛАТФОРМА ДЛЯ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ С ФУНКЦИЕЙ ФОРМИРОВАНИЯ НАВЫКОВ КОМАНДНОЙ РАБОТЫ

С учетом исследованных решений можно сформулировать цель разрабатываемого проекта: создать адаптивную учебную платформу с функцией формирования навыков командной работы. Платформа в формате тематической электронной викторины позволяет оценить и повысить уровень навыков студентов (участников) и группы в целом через решение предъявляемых задач, которые имеют динамически изменяющиеся компоненты и используют механизмы адаптивного обучения. Платформа должна удовлетворять ряду условий и принципов, устраняющих обнаруженные ранее недостатки у прочих платформ подобного рода.

- 1) Модульность – платформа представляет собой систему с модульной архитектурой и иерархией вложенных представлений учебного материала, каждое из которых реализует свой уровень абстракции (курс, лекция, тема, уровень и конкретная задача). Такая гибкая система допускает современный компонентно-ориентированный подход к программной реализации и позволяет создавать нелинейные связи и переходы между отдельными заданиями при построении образовательной траектории группы. Кроме того, появляется возможность создания форм представления заданий для лиц с ограниченными возможностями.
- 2) Микроадаптивность – каждая конкретная задача обладает микроиерархией простейших элементов, реализующих механику для оттачивания конкретного навыка, задействованного в задаче. Эти элементы могут быть также распределены по уровням. Допущенные студентом ошибки при решении участка задачи позволяют определить сегмент знаний, в котором наблюдаются пробелы, а также более тонко корректировать уровень сложности, чтобы участник в ненавязчивой форме мог последовательно отрабатывать и исправлять те ошибки, которые свойственны именно ему.
- 3) Определение входного порога – платформа способна осуществлять адаптивное обучение, подстраиваясь под уровень студента. На входном этапе при регистрации в системе студент проходит адаптивное тестирование, позволяющее в первом приближении оценить уровень навыков. Далее система сама анализирует и корректирует, на основе статистики прохождения этапов викторины, текущий уровень студента.
- 4) Тематическое оформление – задания как таковые представляют собой абстрактные компоненты, которые необходимо заставить корректно работать, чтобы получить требуемую информацию или обеспечить победу в викторине. Для того, чтобы задания можно было использовать в конкретном окружении (тематическом виртуальном пространстве заданий), используются декораторы, которые оформляют механику заданий в форму, привлекательную для участника викторины, используя специальный внутренний декларативный язык программирования,



позволяющий оперативно и эффективно описывать динамически изменяющиеся структуры в элементах предъявляемых задач.

- 5) Навыки командной работы – помимо отдельных заданий, позволяющих установить индивидуальный уровень каждого участника, платформа может оценить способности группы в целом и использовать специальные групповые окружения, где из отдельных компонентов собирается уникальная траектория заданий таким образом, чтобы каждое задание соответствовало уровню выбравшего его участника, при этом результаты отдельных заданий пересекаются, побуждая участников к командному взаимодействию для достижения конечной цели, поставленной групповым окружением. Это позволит не только более точно оценить уровень группы и подбирать более подходящие задания на следующих этапах викторины, но и способствовать формированию навыков командной работы и сплоченности группы в целом (что особенно важно для таких курсов, как, например, программная инженерия).
- 6) Единый дружественный интерфейс для комплекса программ – внешний интерфейс, объединяющий компоненты программного комплекса, обслуживающего адаптивную учебную платформу, должен предоставлять групповую и индивидуальную статистику результатов прохождения этапов викторины в удобной для восприятия преподавателем форме. Впоследствии можно обеспечить и визуальный конструктор или интерактивную среду разработки для упрощения процедуры создания компонентов курса, в частности, учесть специфику разработки и поддержки заданий для лиц с ограниченными возможностями.
- 7) Гибкость реализации – программная реализация платформы должна основываться на принципах переносимости и кроссплатформенности. Необходимо предусмотреть возможность быстрого локального развертывания системы и удобства вызова основных функций, обеспечивающих работу платформы и ее использование для разных программно-аппаратных сред.

КОМПОНЕНТЫ И ОБЩАЯ АРХИТЕКТУРА ЦИФРОВОЙ АДАПТИВНОЙ ПЛАТФОРМЫ

Платформа представляет собой большой программный комплекс, состоящий из не менее трех взаимосвязанных серверов (которые могут быть реализованы с применением технологий виртуализации), не менее двух хранилищ данных и локальной сети, используемой в процессе обучения для осуществления распределенного доступа к рабочей среде и упражнений, формирующих навыки командной работы. Каждый сервер обеспечивает работу набора служб, соответствующих сфере его применения, а также доступ для пользователей, соответствующих роду выполняемых средствами платформы работ (обучение, описание компонентов предъявляемых заданий, сбор и анализ статистики и т.д.). Представление схемы основных компонентов занимает слишком внушительный размер, чтобы ее можно было отобразить в одном рисунке. Общее представление схемы показано на рис. 1, далее будет представлена подробная детализация каждого обобщенного элемента.

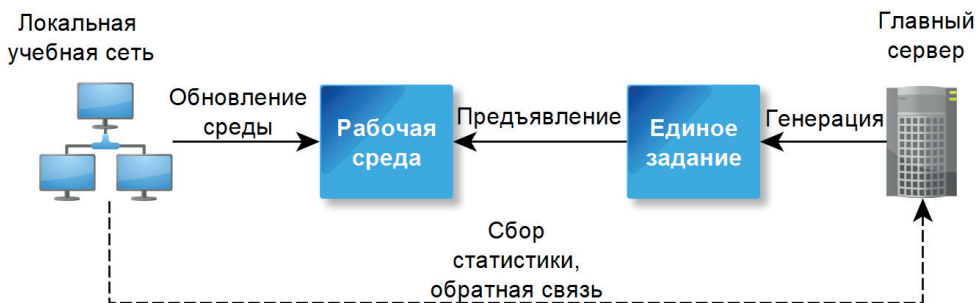


Рис. 1. Общее графическое представление схемы компонентов цифровой адаптивной платформы для обучения программированию с функцией формирования навыков командной работы

Так как фактическая реализация цифровой адаптивной платформы для обучения программированию с функцией формирования навыков командной работы осуществляется в рамках ВУЗа для студентов-бакалавров, далее будет использован термин «студент» для обозначения обучающегося пользователя; однако, стоит отметить, что программный комплекс может осуществлять свои функции и для других образовательных программ (например, в школах).

Набор компонентов, отвечающий за функционирование локальной учебной сети, хранит данные о студентах и группах, которые являются пользователями платформы. Эти данные используются для сопоставления уровня конкретного пользователя и группы в целом, проверки прав доступа на авторизацию и доступ к рабочей среде. Также этот набор компонентов осуществляет функции обновления рабочей среды за счет действий пользователя (например, загрузки валидного сегмента кода, требуемого для прохождения задания) и работу sandbox-сервера, обеспечивающего функции взаимодействия с рабочей средой на стороне клиента с учетом требований информационной безопасности, а также отправку данных на главный сервер о ходе обучения и взаимодействия с рабочей средой – в частности, статистики, необходимой для коррекции уровня подготовки обучающегося и обучения модели машинного обучения, отвечающей за формирование задания и окружения рабочей среды.

Рабочая среда – это виртуальное пространство, обеспечиваемое клиент-серверным взаимодействием между участниками образовательного процесса и главным сервером. Поэтому набор компонентов, обеспечивающий функционирование рабочей среды, носит распределенный характер. Со стороны главного сервера, формирующего рабочую среду, требуется предоставление основы задания, соответствующей текущему сценарию (набору параметров, обеспечивающих взаимосвязь компонентов заданий, и декоративных элементов, поддерживающих вовлеченность и заинтересованность студентов в прохождении заданий). Также на стороне сервера компоненты задания (или динамические составляющие) подбираются с учетом истории взаимодействия участников с рабочей средой и их уровня подготовки. На основе этих данных формируется единое задание и соответствующий ему набор состояний рабочей среды. Со стороны



обучающихся и промежуточных серверов, обслуживающих их взаимодействие с рабочей средой, требуется предоставить функции, обеспечивающие безопасный ввод данных, влияющих на состояние рабочей среды и решение поставленной задачи.

Набор компонентов, обеспечивающих работу главного сервера, представляет собой наиболее высоконагруженную часть всего программного комплекса, так как реализует функции сбора и анализа статистики, процедуры машинного обучения, ответственные за актуализацию учебных сценариев и распределение заданий для студентов и групп по уровням, операции обслуживания и наполнения банка заданий, состоящего из томов баз данных, хранящих сценарии и фрагменты предъявляемых в рабочей среде заданий, а также данные авторизации для лиц, ответственных за ручное наполнение и редактирование банка заданий – например, уполномоченных преподавателей и системных администраторов. Эта же группа лиц имеет доступ к статистике решений по группам и индивидуальной статистике, отражающей прогресс конкретного учащегося лица в изучении предмета средствами платформы. Также главный сервер выполняет свои задачи по формированию единого задания для рабочей среды, учитывающего уровень группы и индивидуальных учащихся.

Набор компонентов, отвечающий за функционирование локальной учебной сети, подробно описан на рис. 2.



Рис. 2. Набор компонентов, соответствующий элементу «Локальная учебная сеть» на общем графическом представлении схемы компонентов

Студенты инициируют работу информационной системы, осуществляя попытку подключения к ее интерфейсу через локальную учебную сеть. Выполняется проверка корректности введенных пользователями данных для входа в систему, для этого применяется сервер групп, который сверяет вводимые данные с зашифрованным содержимым хранилища данных – базы данных о группах; в случае успешной авторизации сервер обеспечивает запуск сессии и допуск для взаимодействия с рабочей средой или обновление информации о текущем состоянии рабочей среды для авторизованных пользователей. Sandbox-сервер подготавливает рабочую среду, которая учитывает характеристики конкретного пользователя (включая его уровень подготовки). Для этого, с помощью полученного на основе данных ав-

торизации идентификатора, sandbox-сервер посылает запрос главному серверу в форме обратной связи и данных статистики о текущем состоянии студента в контексте рабочей среды. Информация, которая поступает в ответ на запрос от главного сервера, позволяет sandbox-серверу сконструировать индивидуальную рабочую среду для каждого студента с учетом его действий и действий прочих студентов в группе. Визуально рабочая среда отображается для студента на стороне клиента – например, браузера, где студент получает возможность осуществлять взаимодействие с рабочей средой в соответствии с предоставляемыми веб-интерфейсом возможностями. Подразумевается, что взаимодействие осуществляется через компьютер в аудитории, имеющий доступ к локальной учебной сети ВУЗа. Однако, принципы клиент-серверной архитектуры позволяют расширить применение программной платформы и для случая дистанционного обучения.

Набор компонентов, отвечающий за функционирование рабочей среды и предъявление единого задания, подробно описан на рис. 3.

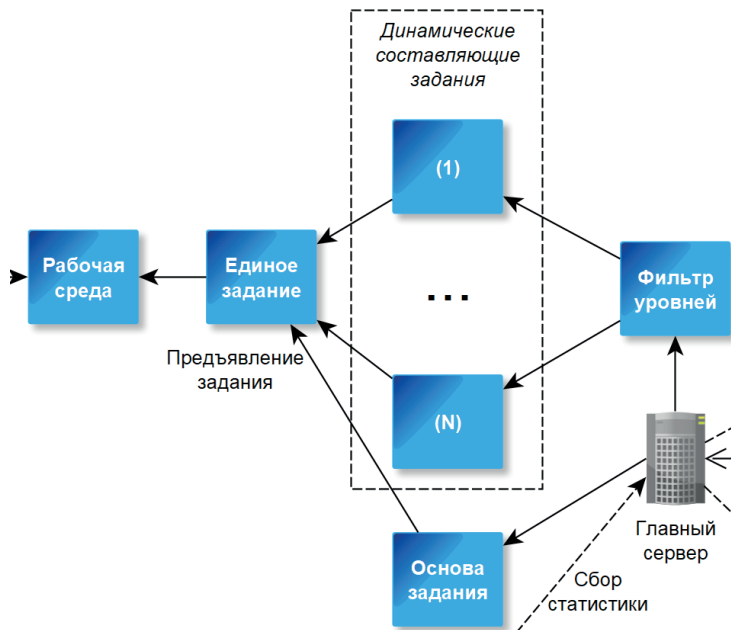


Рис. 3. Набор компонентов, соответствующий элементам «Рабочая среда» и «Единое задание» на общем графическом представлении схемы компонентов

Главный сервер, на основе собранной статистики и полученной обратной связи от sandbox-сервера, подбирает динамические составляющие – компоненты задания – таким образом, чтобы они соответствовали уровню подготовки студента и учитывали ошибки, допущенные в прошлом. На момент, когда тот или иной студент берется за выполнение индивидуального задания, он вкратце знаком только с его основой, описывающей сюжет и тему задания. Когда решение о закреплении индивидуально-



го задания принято, главный сервер, используя данные о студенте и фильтр уровней, модифицирует (подбирает) компоненты задания (детали сюжета и промежуточные задания) таким образом, чтобы они отвечали уровню подготовки студента и позволяли ему проработать допущенные ранее ошибки.

Основа задания задает сюжет для рабочей среды, в котором участвует вся группа. Сюжет и характер задания подбирается исходя из общего уровня группы, показанного на предыдущих занятиях (для первого занятия задается некоторый уровень по умолчанию). После того, как индивидуальные задания выполнены, для достижения общей цели студентам, выполнявшим индивидуальные задания, необходимо, в соответствии с сюжетом, объединить результаты своей индивидуальной работы и решить коллективное задание. Таким образом, формируются навыки командной работы, каждый студент может проявить свои сильные стороны в ходе решения индивидуальных заданий, а также устраняется проблема, при которой один или несколько способных студентов в группе выполняют всю работу за остальных: при наличии индивидуального задания и некоторой части общего для каждого участника не остается ресурсов, чтобы вникнуть и решить чужое индивидуальное задание (с учетом ограничения по времени в 2 академических часа).

После того, как все индивидуальные компоненты постановки задач подобраны и соединены в единое целое с основой задания, задающей сюжет, окружение и групповую задачу, из этих компонентов формулируется единое задание – скрипт, содержащий описание окружения, текстур и прочих декоративных элементов, а также сюжет и основные логические взаимосвязи между результатами индивидуальных заданий и общей для группы цели. На основе этого единого задания формируется рабочая среда и ответы на запросы sandbox-сервера.

Набор компонентов, отвечающий за функционирование главного сервера, подробно описан на рис. 4.

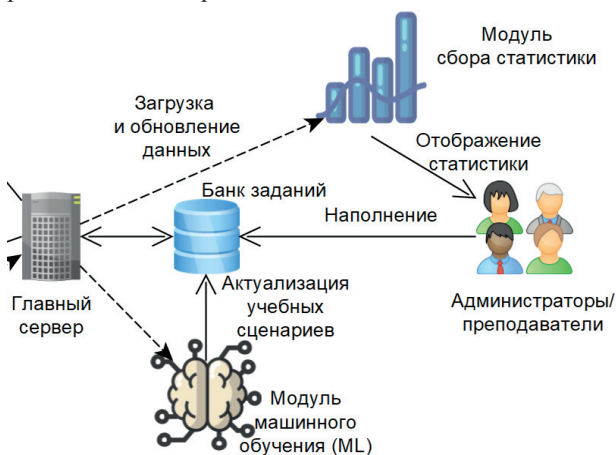


Рис. 4. Набор компонентов, соответствующий элементу «Главный сервер» на общем графическом представлении схемы компонентов

Главный сервер формирует основу задания и подготавливает индивидуальный фильтр уровней для каждого студента, авторизованного и имеющего доступ к учебной среде. На основе этих данных главный сервер выполняет свою часть работы по формированию рабочей среды. Статистические данные, передаваемые в качестве формы обратной связи от sandbox-сервера, главный сервер учитывает, фиксирует и передает в две независимые



службы, реализованные в форме модулей – сбора статистики и машинного обучения. Задача модуля сбора статистики – собрать данные и представить их в наглядной форме уполномоченному лицу, отвечающему за наполнение банка заданий (преподавателю, администратору и т.п.). Модуль машинного обучения использует технологии и модели искусственного интеллекта для коррекции и заполнения банка заданий с целью актуализации учебных сценариев на основе результатов работы студентов, реализуя, тем самым, компонент адаптивности. Лица, отвечающие за наполнение банка заданий, имеют возможность скорректировать результаты работы модуля машинного обучения и осуществлять самостоятельное наполнение банка заданий. Главный сервер также имеет прямой доступ к банку заданий для записи информации, идентифицирующей ход выполнения заданий группой студентов, а также извлечения данных о текущих заданиях и их компонентах для конкретной группы авторизовавшихся студентов.

ЗАКЛЮЧЕНИЕ

В статье были исследованы имеющиеся подходы к построению образовательных сред для обучения программированию. Проанализированы их основные недостатки. Предложена концепция цифровой адаптивной платформы для обучения программированию, учитывающая выявленные недостатки. Описаны особенности программной реализации в виде общей архитектуры, принципов работы и построения взаимосвязей компонентов платформы.

Описанная архитектура реализует все условия и принципы, позволяющие преодолеть выявленные у похожих продуктов недостатки:

- 1) модульность и микроадаптивность – реализованы за счет системы динамических составляющих задания;
- 2) определение входного порога – за счет модуля сбора статистики и функций sandbox-сервера;
- 3) тематическое оформление и единый дружественный интерфейс для комплекса программ – за счет модулей и функций главного сервера;
- 4) навыки командной работы – за счет создания виртуальной рабочей среды на базе клиент-серверной архитектуры и предъявления единого задания, учитывающего уровень подготовки группы и совокупность результатов выполнения индивидуальных заданий;
- 5) гибкость реализации – за счет компонентного подхода к построению цифровой адаптивной платформы для обучения программированию с функцией формирования навыков командной работы.

Таким образом, в статье были продемонстрированы особенности программной реализации цифровой адаптивной платформы для обучения программированию с функцией формирования навыков командной работы.

Литература

1. Кодвардс – URL: <https://codewards.ru> (дата обращения 10.12.2020).
2. CoLoBot for Windows (2001) – MobyGames – URL: <https://www.mobygames.com/game/colobot> (дата обращения 10.12.2020).



3. Гудфеллоу Я., Бенджио И., Курвилль А. Глубокое обучение / пер. с англ. А.А. Слинкина. – М.: ДМК Пресс, 2017. – 652 с.: ил.
4. Kuravsky L.S., Popkov S.I. and Artemenkov S.L. An applied multi-agent system within the framework of a player-centered probabilistic computer game. – International Journal of Modeling, Simulation, and Scientific Computing, Vol. 9, No. 1 (2018), 17 pp, DOI: 10.1142/S1793962317500635 [SCOPUS].
5. Kuravsky L.S., Popkov S.I. and Artemenkov S.L. Applied multi-agent system to study behavior of operators of complex technical systems. – In: Proc. First World Congress on Condition Monitoring 2017 (WCCM 2017) The International Society for Condition Monitoring (ISCM), British Institute of Non-Destructive Testing (BINDT). 2017 [SCOPUS].
6. Moodle – URL: <https://moodle.org> (дата обращения 10.12.2020).
7. edX – URL: <https://www.edx.org> (дата обращения 10.12.2020).
8. Chamilo LMS – URL: <https://chamilo.org/en/> (дата обращения 10.12.2020).
9. FlexiQuiz – URL: <https://www.flexiquiz.com> (дата обращения 10.12.2020).
10. ClassMaker – URL: <https://www.classmarker.com> (дата обращения 10.12.2020).
11. Pesofts – URL: <https://pesofts.com> (дата обращения 10.12.2020).
12. SberCraft – URL: <https://sber.geecko.ru> (дата обращения 10.12.2020).
13. Redmadrobot запускает платформу «Кодвардс» для обучения детей основам программирования – URL: <https://www.redmadrobot.ru/fyi/codewards> (дата обращения 10.12.2020).
14. CodinGame – URL: <https://www.codingame.com/start> (дата обращения 10.12.2020).
15. CodinGame for Work – URL: <https://www.codingame.com/work> (дата обращения 10.12.2020).
16. Official Colobot: Gold Edition website – URL: <https://colobot.info> (дата обращения 10.12.2020).



Features of the Software Implementation of the Digital Adaptive Platform for Learning Programming with the Teamwork Skills Forming Function

Sergei I. Popkov*

Moscow State University of Psychology and Education, Moscow, Russia

ORCID: <https://orcid.org/0000-0003-2566-1262>

e-mail: rslw25@gmail.com

A study of existing approaches to creating educational environments for learning programming is conducted. Their main shortcomings are analyzed. The concept of a digital adaptive platform for learning programming, taking into account the identified shortcomings, is proposed. The features of the software implementation are described in a form of links between platform components, operation concepts and the platform architecture overall. The features of the software architecture and implementation of the digital adaptive platform for learning programming with the teamwork skills forming function are demonstrated.

Keywords: adaptability, digital platform, learning computer programming, teamwork skills.

For citation:

Popkov S.I. Features of the Software Implementation of the Digital Adaptive Platform for Learning Programming with the Teamwork Skills Forming Function. *Modelirovanie i analiz dannyykh = Modelling and Data Analysis*, 2020. Vol. 10, no. 3, pp. 85–100. DOI: <https://doi.org/10.17759/mda.2020100307> (In Russ., abstr. in Engl.).

References

1. Codewards – URL: <https://codewards.ru> (req. date 10/12/2020)/
2. CoLoBot for Windows (2001) – MobyGames – URL: <https://www.mobygames.com/game/colobot> (req. date 10/12/2020)/
3. Ian Goodfellow, Yoshua Bengio, Aaron Courville. Deep Learning (in Russ.) – M.: DMK Press, 2017. – 652 pp.: il.
4. Kuravsky L.S., Popkov S.I. and Artemenkov S.L. An applied multi-agent system within the framework of a player-centered probabilistic computer game. – *International Journal of Modeling, Simulation, and Scientific Computing*, Vol. 9, No. 1 (2018), 17 pp, DOI: 10.1142/S1793962317500635 [SCOPUS].
5. Kuravsky L.S., Popkov S.I. and Artemenkov S.L. Applied multi-agent system to study behavior of operators of complex technical systems. – In: Proc. First World Congress on Condition Monitoring 2017 (WCCM 2017) The International Society for Condition Monitoring (ISCM), British Institute of Non-Destructive Testing (BINDT). 2017 [SCOPUS].

***Sergei I. Popkov**, PhD in Physical and Mathematical Sciences, Assistant Professor of the faculty of information technologies, head of the laboratory, Moscow State University of Psychology and Education, Moscow, Russia, ORCID: <https://orcid.org/0000-0003-2566-1262>, e-mail: rslw25@gmail.com



6. Moodle – URL: <https://moodle.org> (req. date 10/12/2020).
7. edX – URL: <https://www.edx.org> (req. date 10/12/2020).
8. Chamilo LMS – URL: <https://chamilo.org/en/> (req. date 10/12/2020).
9. FlexiQuiz – URL: <https://www.flexiquiz.com> (req. date 10/12/2020).
10. ClassMaker – URL: <https://www.classmarker.com> (req. date 10/12/2020).
11. Pesofts – URL: <https://pesofts.com> (req. date 10/12/2020).
12. SberCraft – URL: <https://sber.geecko.ru> (req. date 10/12/2020).
13. “Redmadrobot” company launches «Codewards» platform to teach children the basics of the programming – URL: <https://www.redmadrobot.ru/fyi/codewards> (req. date 10/12/2020).
14. CodinGame – URL: <https://www.codingame.com/start> (req. date 10/12/2020).
15. CodinGame for Work – URL: <https://www.codingame.com/work> (req. date 10/12/2020).
16. Official Colobot: Gold Edition website – URL: <https://colobot.info> (req. date 10/12/2020).

Моделирование и анализ данных 2020. Том 10. № 3.

Научный журнал

Издаётся с 2011 года

Учредитель

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Московский государственный психолого-педагогический университет»

Адрес редколлегии:

г. Москва, ул. Сретенка, 29, факультет информационных технологий

Тел.: +7 (499) 167-66-74

E-mail: mad.mgppu@gmail.com

Журнал зарегистрирован в Государственном комитете РФ по печати.

Свидетельство о регистрации средств массовой информации

ПИ № ФС77-52058 от 7 декабря 2012 года

ISSN: 2219-3758

ISSN: 2311-9454 (online)

Подписано в печать: 28.12.2020.

Формат: 70*90/16. Бумага офсетная.

Гарнитура Times. Печать цифровая.

Усл. печ. п. 6,3. Усл.-изд. л. 8,7.

Тираж 500 экз.