ISSN 1818–1015 (Print)
ISSN 2313–5417 (Online)

Министерство образования и науки Российской Федерации
Ярославский государственный университет им. П. Г. Демидова

МОДЕЛИРОВАНИЕ И АНАЛИЗ ИНФОРМАЦИОННЫХ СИСТЕМ

Том 24 № 2(68) 2017

Основан в 1999 году
Выходит 6 раз в год

Адрес редакции: ЯрГУ, ул. Советская, 14, г. Ярославль, 150003, Россия
Website: http://mais-journal.ru, e-mail: mais@uniyar.ac.ru; телефон (4852) 79-77-73

Научные статьи в журнал принимаются по электронной почте. Статьи должны содержать УДК, аннотации на русском и английском языках и сопровождаться набором текста в редакторе LaTeX. Плата с аспирантов за публикацию рукописей не взимается.

12+

СОДЕРЖАНИЕ

Моделирование и анализ информационных систем. Т. 24, №2. 2017

Свидетельство о регистрации СМИ ПИ № ФС 77 – 66186 от 20.06.2016 выдано Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций. Учредитель – Федеральное государственное бюджетное образовательное учреждение высшего образования "Ярославский государственный университет им. П. Г. Демидова". Подписной индекс – 31907 в Объединенном каталоге "Пресса России". Редактор, корректор А.А. Аладьева. Редактор перевода Э.И. Соколова. Подписано в печать 18.04.2017. Дата выхода в свет 30.04.2017. Формат 60x84$^1/_8$. Усл. печ. л. 15,34. Уч.-изд. л. 13. Объем 132 с. Тираж 46 экз. Свободная цена. Заказ 016/017. Адрес типографии: ул. Советская, 14, оф. 109, г. Ярославль, 150003 Россия. Адрес издателя: Ярославский государственный университет им. П. Г. Демидова, ул. Советская, 14, г. Ярославль, 150003 Россия.

ISSN 1818–1015 (Print)
ISSN 2313–5417 (Online)

P.G. Demidov Yaroslavl State University

MODELING AND ANALYSIS OF INFORMATION SYSTEMS

Volume 24 No 2(68) 2017

Founded in 1999
6 issues per year

Editorial Office Address: P.G. Demidov Yaroslavl State University,
14 Sovetskaya str., Yaroslavl 150003, Russia
Website: http://mais-journal.ru, e-mail: mais@uniyar.ac.ru

Contents

Modeling and Analysis of Information Systems. Vol. 24, No 2. 2017

Моделирование и анализ информационных систем. Т. 24, № 2 (2017), с. 125–140
Modeling and Analysis of Information Systems. Vol. 24, No 2 (2017), pp. 125–140

DOI: 10.18255/1818-1015-2017-2-125-140

UDC 517.9

Discovering High-Level Process Models from Event Logs

Begicheva A. K., Lomazova I. A.

Received January 11, 2017

Abstract. Process mining is a relatively new field of computer science, which deals with process discovery and analysis based on event logs. In this paper we consider the problem of discovering a high-level business process model from a low-level event log, i.e. automatic synthesis of process models based on the information stored in event logs of information systems. Events in a high-level model are abstract events, which can be refined to low-level subprocesses, whose behavior is recorded in event logs. Models synthesis is intensively studied in the frame of process mining research, but only models and event logs of the same granularity are mainly considered in the literature. Here we present an algorithm for discovering high-level acyclic process models from event logs and some specified partition of low-level events into subsets associated with abstract events in a high-level model.

Keywords: Petri nets, high-level process models, event logs, process mining, process discovery

For citation: Begicheva A. K., Lomazova I. A., "Discovering High-Level Process Models from Event Logs", *Modeling and Analysis of Information Systems*, **24**:2 (2017), 125–140.

About the authors:
Antonina A. Begicheva, research assistant
National Research University Higher School of Economics, Laboratory of Process-Aware Information Systems
20 Myasnitskaya str., Moscow 101000, Russia, e-mail: akbegicheva@edu.hse.ru

Irina A. Lomazova, Doctor of science, professor
National Research University Higher School of Economics
20 Myasnitskaya str., Moscow 101000, Russia, e-mail: ilomazova@hse.ru

Acknowledgments:
This work is supported by the Basic Research Program at the National Research University Higher School of Economics and Russian Foundation for Basic Research, project No.16-01-00546.

1. Introduction

Process mining is a technology that provides a variety of methods to discover, monitor and improve real processes by extracting knowledge from event logs [1]. Process discovery and conformance checking are the two most prominent process mining tasks. Process discovery is needed to construct a process model, based on an event log, without any additional information. Conformance checking helps us in diagnosing and quantifying discrepancies between observed and modeled behavior. Process discovery uses only an event log for recovery view of a system as a model by behavior which is seen in the log. The general goal is to find out main events of the system and relations between them.

After this we can use other techniques of process mining for further work with the model of our system.

There are many software products which allow us to use these two techniques of Process Mining. ProM [2] is an open-source tool supporting many techniques of Process Mining, which are represented as plug-ins. Due to the flexibility of this environment, it can be used both for research and applications.

When working with business processes we typically use detailed logs, which present the full report about sequences of executed activities. Since logs are generated automatically in most information systems, keeping detailed records is not a problem. However, large and detailed models are not good to deal with. Such models are not clear and readable for experts. Experts prefer working with more abstract (high-level) models. More abstract models are easier to construct, understand and analyze. Process models developed by people are, as a rule, not very large and abstract from technical details. Although the field of process discovery is well developed, process mining projects still face the problem of different levels of abstraction when comparing events with modeled business activities. Current approaches for event log abstraction most often try to abstract from the events in an automated way which does not capture the required domain knowledge to fit business process.

In this work we consider process models represented by workflow nets – a special class of Petri nets [3] for workflow modeling. We assume that a model contains no cycles for correct handling with concurrency. In an abstract model each separate activity represents a subprocess built from a set of more refined activities. For the presentation of intermediate results we use transition systems. A history of a detailed process behavior is recorded in low-level logs. We present an algorithm for discovering process as an abstract model from a low-level event log. The algorithm will be tested on groups of input data with different characteristics.

The work is organized as follows. Section 2 introduces the situation with existing researches. In Section 3 we give a motivating example of handling a request for compensation in a particular airline company, in terms of Petri nets. Section 4 contains some basic definitions and notions, including Petri nets, event log, transition system, and theory of regions. In Section 5 we present a method for discovering an abstract process model from a low-level event log, and Section 6 gives a precise description and validation of this method. Section 7 concludes the paper.

2. Literature review

Research topics related to this article can be divided into several categories: event log and model abstraction, discovering algorithms and existing methods for abstract models synthesis.

There are many ways of abstracting models by reducing their size in order to make them more convenient to work with. Each method may be useful depending on a group of interrelated factors: the abstraction purposes, the presence of certain patterns of routine constructions, the specifics of modeling notation. Reducing the size of the model by abstraction can be done as “convolution” of some groups of elements, or implemented by losing some parts (which are non-significant in a particular case). The importance of event

log abstraction is emphasized among others in [4]. The paper contains a more detailed overview of models and abstraction techniques and formal definitions of this process in general. In particular, the paper provides definitions for two prominent abstraction operations: *elimination* and *aggregation*. A model which is generated by an elimination operation contains no information about omitted insignificant objects. In contrast the aggregation generates an abstract model, where relatively insignificant objects of abstraction combine together with several other abstraction objects into groups, each of which is significant. According to this article, business process model abstraction is an operation over the model, which transforms it into abstract model by the application of function composition, where each component of the composition is some basic abstract operation.

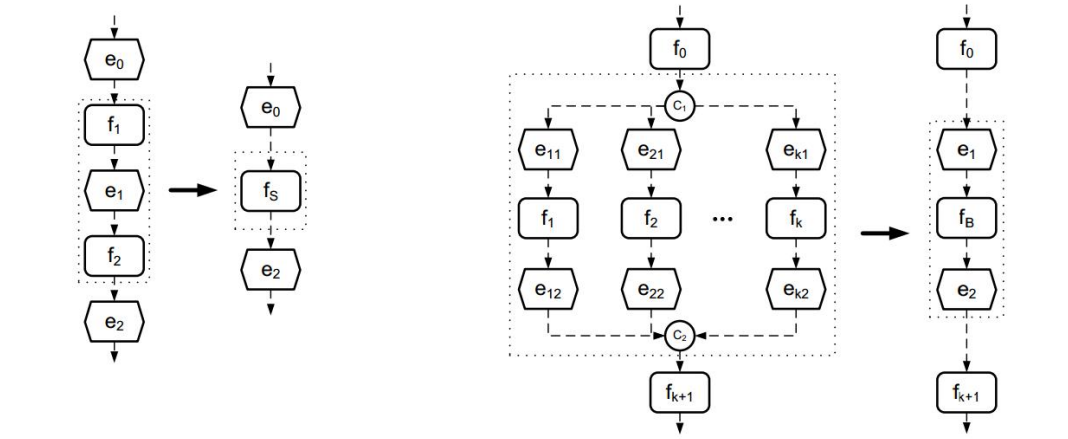
The aggregation is divided into four types: sequential abstraction (Fig. 1 (a)), block abstraction (Fig. 1 (b)), loop abstraction (Fig. 1. (c)), dead end abstraction (Fig. 1. (d)). Based on these four aggregation rules and model transformation techniques a slider approach is proposed in [5]. It provides a user control over the level of model abstraction by different criteria, where the abstraction criterion is a pair: type of criteria and relation between element property and its significance. A similar ability is used in many methods of working with abstract models due to its convenience and an analogy with scalable terrain maps.

Petri nets can be extended by adding a hierarchy e.g. in Colored Petri nets (CPN) [6]. The idea of high-level Petri nets was first described in [7] (1981), but only in 1987 this idea extended to colored Petri nets. Hierarchy also allows construction of more compact, readable and understandable models. Hierarchy can be applied as an abstraction, in the case of two-level hierarchy there are two models of one process: a high-level *abstract* model and a low-level *refined* model. In our paper the high-level model is a model with abstract transitions. An abstract transition refers to a Petri net subprocess, which refines the activity represented by this transition. The low-level model can be obtained from an abstract model by substituting subprocess models for abstract transitions.

A "flat" synthesis (when model and log are at the same-level) is a popular process mining task, and has been extensively studied in the literature. Each method [1, 8, 9] has some advantages regarding a particular kind of data. For example the α-algorithm [9] takes concurrency as a base, but this algorithm has some problems dealing with complicated routing constructs and noise. The flat model, presented as a graph, becomes more incomprehensible with each node. Hierarchical models are more suitable for humans to work with because they are more structural.

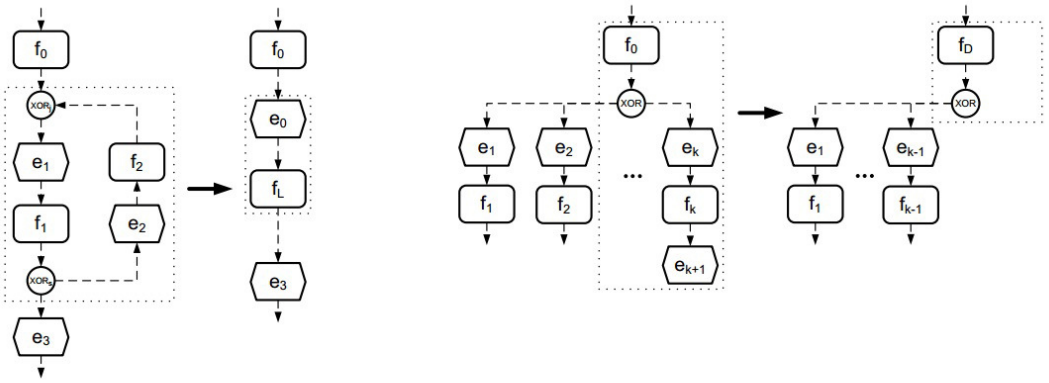
Van Dongen et al in [10] described a Petri net discovery approach using transition system. Firstly a transition system is constructed by deriving from an event log. It can be modified to avoid over-fitting. Secondly, using the theory of regions [11], the Petri net is synthesized. We note that currently only the theory of regions solves the problem of Petri net synthesis from transition systems. The proposed in [10] method can deal with complex control-flow constructs. It allows for duplicates, but does not allow for much more behavior than is actually recorded in the log and produces models, which satisfy some soundness requirements. The proposed algorithm also has some disadvantages, for example it can't deal with noise in an event log, unlike genetic [12] or heuristics [13] miners.

There are some methods for discovering abstract models but they are based on finding behavior patterns in event logs. For example in [14] the authors use recognition of



(a) Sequential abstraction

(b) Block abstraction



(c) Loop abstraction

(d) Dead end abstraction

Fig 1. Aggregations for model abstraction

behavior patterns of the process by a structural clustering algorithm and then define a specific workflow schema for each pattern. This helps to divide one complex model into several more understandable, smaller pieces, convenient to analyze and discover. Clustering of activities by their relation and role in the process was also used in [15]. The terrain maps analogy is also used here: activities in a process correspond to a certain location on a map and the relations between activities are like the roads.

In [16] a supervised event abstraction method was presented. This method takes an event log at a lower level of abstraction and transforms it to an event log at the desired level of abstraction, using behavioral activity patterns: sequence, choice, parallel, interleaving and repetition. This technique allows us to obtain a reliable abstraction mapping from low-level events to activity patterns automatically and construct a high-level event log using them. Another supervised event abstraction method is described in [17]. The essence of the proposed method is as follows: we annotated each low-level event with the correct high-level event using domain knowledge from the actual process model by special type of attribute in XES log file. Also in this paper authors make the working assumption that multiple high-level events are executed in parallel. This enables us to interpret a sequence of identical label attribute values as a single instance of a high-level event.

In [18] a two-phase approach to mining process is presented. Process models here are considered as interactive and context-dependent maps based on common execution patterns. On the first phase the event log is transformed to the desired level of detail by selecting patterns. An example of such pattern is the maximal repeat, that captures typical sequences of activities the log. Every pattern is estimated by frequency, or significance, or some other metric needed for accurate abstraction. At the second phase the fuzzy miner algorithm adapted to process maps discovery is applied to the transformed log. This two-phase approach has been implemented as a set of interrelated plug-ins in the ProM framework, the application order of which is described in [19]. In our paper we use Petrify plug-in from this set to synthesize Petri net [20].

All these papers provide no or only limited support for correct refining these mappings based on domain knowledge. They do not allow to detect subprocesses in the synthesized model. Some approaches based on subprocess detection for mining a process model with a better structure were presented in [21, 22], but these papers do not consider mining high-level models.

Also there are different approaches that apply behavior abstraction in process discovery and trace alignment [5, 23]. Besides there is an interesting method for discovering that is discussed in [24]. The contribution of this research is a mapping approach which suggests relations between events and activities in an automated manner using existing process documentation as e.g. work instructions.

Thus discovering an abstract model from a low-level event log generated by an information system is an important and challenging problem. In different applications different views on the abstraction mechanism and different levels of abstraction are needed. Here we consider one of many possible views on the problem.

$$
\begin{aligned}
L = \{ &< t0, t1, t3, t4, t5, t6, t13, t14, t7, t8, t15, t9, t16, t17, t18, t25, t26, t27 >, \\
&< t0, t1, t3, t4, t13, t14, t5, t7, t15, t6, t8, t9, t16, t17, t18, t24, t23, t22 >, \\
&< t0, t1, t3, t13, t10, t11, t14, t15, t12, t16, t17, t18, t24, t23, t22 >, \\
&< t0, t2, t3, t13, t4, t14, t15, t5, t6, t7, t8, t9, t16, t17, t18, t25, t26, t27 >, \\
&< t0, t2, t3, t4, t13, t14, t15, t5, t7, t6, t8, t9, t16, t17, t18, t24, t23, t22 >, \\
&< t0, t1, t3, t10, t11, t13, t12, t14, t15, t16, t17, t18, t19, t20, t21, t10, t11, \\
&t13, t14, t15, t12, t16, t17, t18, t25, t26, t27 >, \\
&< t0, t2, t3, t13, t10, t14, t15, t11, t12, t16, t17, t18, t24, t23, t22 >, \\
&< t0, t2, t3, t13, t10, t11, t12, t14, t15, t16, t17, t18, t19, t20, t21, t10, t13, \\
&t14, t11, t15, t12, t16, t17, t18, t24, t23, t22 >, \\
&< t0, t2, t3, t13, t14, t10, t11, t15, t12, t16, t17, t18, t25, t26, t27 >\}.
\end{aligned}
$$

Fig 2. An original event log $\mathcal{L}_1$

3. Motivation example

Let us consider some log, which describes in detail a half-hour period of handling a request for a compensation from the airline service. The log is generated by a system, a part of this los is shown in Fig. 2. Names of events are simplified to t with some index, because with real names the log was too bulky. Even if names of events have not been replaced, it is not easy to understand from the log how customers requests are handled. But there are algorithms allowing to discover (synthesize) a process model from this log.

After applying one of the known discovery algorithms we get the relatively large model $\mathcal{N}_1$ presented in Fig. 3. This model is inconvenient to work with. Experts would prefer to work with more abstract model of the same process, like the model $\mathcal{N}_2$ shown in Fig. 4. Each transition in $\mathcal{N}_2$ corresponds to a subprocess, which includes transitions from $\mathcal{N}_1$. For example, the abstract action 'register request' indicates reading a request (transition $t0$) and then recording it in one of two possible ways (transition $t1$, or $t2$). Note that sets of low-level transitons corresponding to different abstract events do not intersect. Low-level transitions in Fig. 3 are grouped into blocks (subprocesses) corresponding to

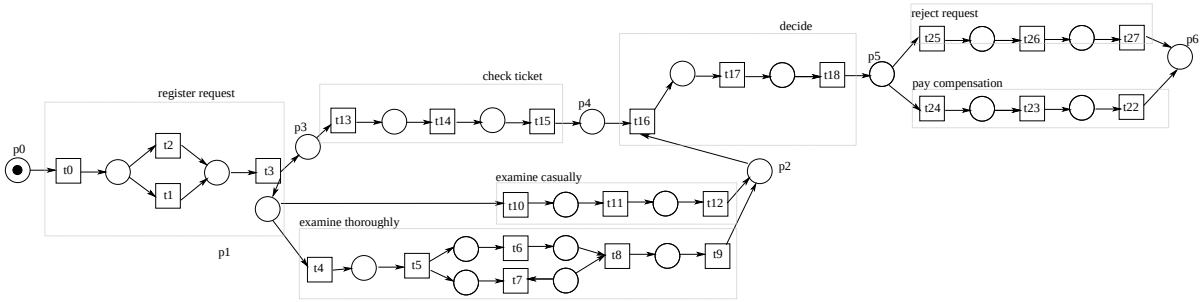


Fig 3. A low-level model $\mathcal{N}_1$ synthesized from the log $\mathcal{L}_1$ in Fig. 2

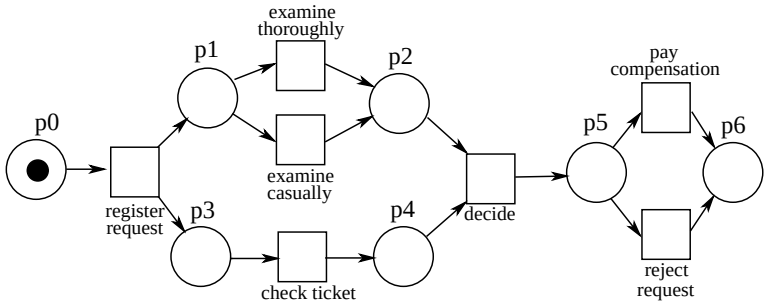


Fig 4. An abstract model $\mathcal{N}_2$ for handling compensation requests

high-level activities of the process.

We study the problem when given a low-level event log generated by a process and a partition of the set of low-level events into subsets corresponding to high-level events , we would like to construct an abstract model of the process. We suppose that the low-level log is generated by an information system, and the event partition is defined by experts or software developers

An abstract model cannot be directly obtained from low-level log since the log consists of low-level events and the model should contain transitions labeled by high-level events. So, a one-to-many correspondence between high and low-level events is needed for the model synthesis. In our example the abstract event $e0$ corresponds to the set $\{t0, t1, t2\}$ of low-level events, $e1$ — to the set $\{t13, t14, t15\}$, etc. The full correspondence is shown in Table 1. This mapping will be used in the algorithm for transformation of the original log into its high-level representation, which we use as an input data for one of the known discovery algorithm.

High-Level Activity	e0	e1	e2	e3	e4	e5	e6	e7
Low-Level Activities	t0	t4	t10	t13	t16	t19	t22	t25
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	t3	t9	t12	t15	t18	t21	t24	t27

Table 1. The correspondence between low-level and high-level activities in $\mathcal{N}_1$ and $\mathcal{N}_2$

4. Preliminaries

In this section we give some basic notions and definitions used in the paper.

Let S be a set. By S^* we denote the set of all finite sequence (words) over S.

$S = S_1 \cup S_2 \cup \cdots \cup S_n$ is a partition of S iff $\forall i, j \in [1, n] : S_i \subseteq S$ and $S_i \cap S_j = \emptyset$.

A multiset m over a set S is a mapping: $m : S \to Nat$, where Nat – is the set of natural numbers (including zero), i.e. a multiset may contain several copies of the same element.

Definition 1 (Petri net)**.** *Let P and T be disjoint finite sets of places and transitions and $F : (P \times T) \cup (T \times P) \to Nat$. Then $N = (P, T, F)$ is a* Petri net. *Let A be a finite set*

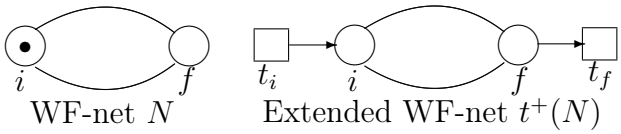


Fig 5. Extending WF-net with initial and final transitions

of activities. A labeled *Petri net is a Petri net with a labeling function* $\lambda : T \to A \cup \{\epsilon\}$, *which maps every transition to an activity (a transition label) from* A, *or a special label* ϵ *indicating an invisible action.*

A marking in a Petri net is a function $m : P \to Nat$ mapping each place to some natural number (possibly zero).

For a transition $t \in T$ a *preset* ${}^\bullet t$ and a *postset* $t^\bullet$ are defined as the multisets over P such that ${}^\bullet t = \{p | F(p,t) \neq 0\}$ and $t^\bullet = \{p | F(t,p) \neq 0\}$.

A transition $t \in T$ is *enabled* in a marking m iff $\forall p \in P\ m(p) \geq F(p,t)$. An enabled transition t may *fire* yielding a new marking m', such that $m'(p) = m(p) - F(p,t) + F(t,p)$ for each $p \in P$ (denoted $m \xrightarrow{t} m'$).

A *Workflow net* is a (labeled) Petri net with two special places: i and f. These places are used to mark the beginning and the ending of a workflow process.

Definition 2 (Workflow net)**.** *A (labeled) Petri net* $\mathrm{N} = (\mathrm{P}, \mathrm{T}, \mathrm{F}, \lambda)$ *is called a* workflow net *(*WF-net*) iff:*

1. *There is one source place* $i \in P$ *and one sink place* $f \in P$, *s. t.* ${}^\bullet i = f^\bullet = \emptyset$.
2. *Every node from* $P \cup T$ *is on a path from* i *to* f.
3. *The initial marking in* N *contains the only token in its source place.*

By abuse of notation we denote by i both the source place and the initial marking in a WF-net. Similarly, we use f to denote the final marking in a WF-net N, defined as a marking containing the only token in the sink place f.

Let $N = (P, T, F, \lambda)$ be a WF-net. Then we define the extended WF-net (EWF-net) $N' = (P', T', F', \lambda')$ as follows: $P' = P, T' = T \cup \{t_i, t_f\}$ and $F' = F \cup \{\langle t_i, i\rangle, \langle f, t_f\rangle\}$, where t_i, t_f are new (not occurring in P, T) nodes. The new transitions t_i, t_f are labeled with invisible activity ϵ, all other transitions in N' have the same labels as in N. The initial marking in an extended WF-net contains no tokens. Thus an extended WF-net may start a new case at any moment (cf. Fig. 5).

The behavior of WF-nets can be represented with a state-based models, called Transition System (TS), as they reflect the states of a process and transitions between them.

Definition 3 (Transition system)**.** *A (labeled) transition system is a triple* $(S, \Lambda, \to)$, *where* S *is a set of states,* Λ *is a set of labels, and* $\to \subseteq S \times \Lambda \times S$ *is a transition relation. If* $p, q \in S$ *and* $\alpha \in \Lambda$, $(p, \alpha, q) \in \to$ *is usually written as* $p \xrightarrow{\alpha} q$ *meaning that a transition labeled by* α*moves the system from state* p *to state* q. *Furthermore, in this paper we assume that a transition system is connected.*

Information systems can record all kinds of events with a wide range of properties, but it entirely depends on the configuration. Moreover, systems can use a specific format. For our study we abstract from additional information presented in event logs and define a *process log* as a multiset of *traces*, where a *trace* is a sequence of *events* (only their names).

In this paper a *path* is a sequence of events apart from the log, in the context of a one of possible runs from initial state of a model to final state.

Definition 4 (Event log)**.** *Let A be a finite set of activities. A trace σ is a finite sequence of activities from A, i.e. $\sigma \in A^*$. An event log L is a finite multi-set of traces, i.e. $L \in \mathcal{M}(A^*)$.*

A model represented as transition system TS corresponding to a model represented as Petri net PN iff TS can be successfully run with all possible paths of TS and vice versa.

To apply the Petri net synthesis method, we need to transform event logs into transition systems [10]. Using the translation from a single trace to a transition system, we can translate an entire log to a transition system.

Definition 5 (Event log to transition system)**.** *Let A be a set of log events and let W be an event log over A, i.e., $W \in \mathcal{M}(A^*)$. We define $TS(W) = (S; \Lambda; \rightarrow)$ to be a transition system, such that:*

- $S = \bigcup_{\sigma \in W} S_\sigma$*, i.e. the union over the states of the transition system translations of each individual trace,*
- $\Lambda = A$*, i.e. the set of labels is the set of activities,*
- $\rightarrow = \bigcup_{\sigma \in W} \rightarrow_\sigma$*, i.e. the trace is represented as a sequence of state transitions, starting from the common initial state. The transitions between each two states is made by activity at the given position in the trace.*

The algorithm [8] for constructing a transition system is straightforward: for every trace σ, iterating over $k (0 \leq k \leq |\sigma|)$, we create a new state $state(\sigma, k)$ if it does not exist yet. Then the traces are scanned for transitions $state(\sigma, k-1) \xrightarrow{\sigma(k)} state(\sigma, k)$ and these are added if it does not exist yet. If we use the complete prefix sequence representation of a state, i.e. $state(\sigma, k) = hd(\sigma, k)$, where $hd(\sigma, k)$ is a *prefix* of trace σ after executing k steps.

Once a process log is converted into transition system, we can use the some synthesis method to generate a Petri net from it, at the time of writing this article for these purposes, you can apply only Theory of Regions.

For synthesis we use Theory of Regions, and to able using it we need to make an assumption about completeness of the log. We assume also that the log shows all possible behavior, since the resulting Petri net have to be exactly mimic the behavior shown in the log. Since we work only with workflow process, we do not need to make any assumptions about the uniqueness of initial state for each trace. More details about the application of the theory of regions can be found in [10, 25].

The advantages of using theory of regions are the possibility to handle complex models with routing constructions (such as concurrency), the possibility of deduplication in the

construction of the resulting net, the flexibility, due to which the model does not allow too general behavior and are not too "tight" for some concrete routing. The disadvantage of theory of regions is high computational complexity, which makes it difficult to use for synthesis models from large size logs. The model abstraction allow us to use all of advantages of theory of regions faced with no disadvantages.

Second of the most prominent process mining tasks is conformance checking [1, 26, 27]. Given a model and an event log we would like to compare the process model behavior and the behavior recorded in the event log. Several metrics for conformance checking were defined in the literature [1]. Among the most important metrics is *fitness*. Informally speaking, fitness measures the proportion of behavior in the event log possible according to the model.

Definition 6 (Perfect fit)**.** *Let N be a WF-net with transition labels from A, an initial marking i, and a final marking f. Let σ be a trace over A. We say that a trace $\sigma = a_1, \ldots, a_k$* perfectly fits *$N$ iff there exists a sequence of firings $i = m_0 \xrightarrow{t_1} \ldots \xrightarrow{t_k} m_{k+1} = f$ in N, s.t. the sequence of activities $\lambda(t_1), \lambda(t_2), \ldots, \lambda(t_k)$ after deleting all invisible activities ϵ coincides with σ. A log L* perfectly fits *N iff every trace from L perfectly fits N.*

Workflows can be modeled with varying degrees of detailing. This idea is realized in models of different abstraction levels. The correspondence between models of different level, which describe the same process, is natural to match using refinements. When a transition from the model is a reference to the sub process, this transition is an abstract (high-level) one. In this case, we can obtain detailed (low-level) model by substituting of corresponding sub-process model instead of high-level transition. This hierarchy principle is used, for example, in colored Petri nets (CPN) [6]. Refinements allow us to develop a more compact model with the composite structure of the network. Here we give precise definitions introduced in [27].

Definition 7 (Substitution)**.** *Let $N_1 = (P_1, T_1, F_1, \lambda_1)$ be a WF-net, $t \in T$ be a transition in N_1. Let also $N_2 = (P_2, T_2, F_2, \lambda_2)$ be an EWF-net with the initial and final transitions t_i, t_f correspondingly. We say that a WF-net $N_3 = (P_3, T_3, F_3, \lambda)$ is obtained by a* substitution *$[t \to N_2]$ of N_2 for t in N_1 iff $P_3 = P_1 \cup P_2$, $T_3 = T_1 \cup T_2 \setminus \{t\}$, $F_3 = F_1 \cup F_2 \setminus \{(p,t) \mid p \in {}^\bullet t\} \setminus \{(t,p) \mid p \in t^\bullet\} \cup \{(p, t_i) \mid p \in {}^\bullet t\} \cup \{(t_f, p) \mid p \in t^\bullet\}$,*

Definition 8 (Refinement)**.** *Let N, N_r be two WF-nets with sets of activities A, A_r correspondingly. Let $A = a_1, a_2, \ldots, a_n$, and $A_r = A_r^1 \cup A_r^2 \cup \ldots \cup A_r^n$ be a partition of A_r into n subsets, and $N^1, N^2, \ldots N^n$ be EWF-nets with sets of activities $A_r^1, \ldots, A_r^n$ correspondingly. We say that N_r is a refinement of N via substitutions $[a_1 \to N_r^1, a_2 \to N_r^2, \ldots a_n \to N_r^n]$ iff N_r can be obtained from N by simultaneous substitutions of N_r^i for all t s.t. $\lambda(t) = a_i$.*

5. Synthesis of Abstract Process Model from a Low-Level Event Log

In this section we describe, how to obtain an abstract model from a given low-level log L. We suppose that the set A_r of low-level events in the log L is partitioned into subsets

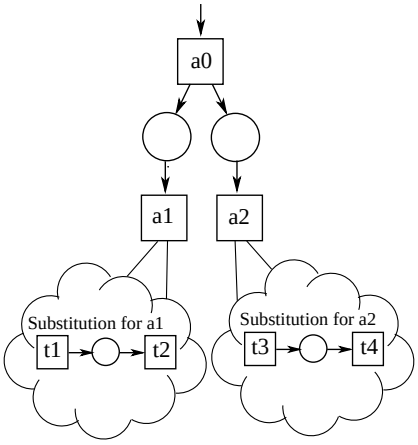


Fig 6. Concurrent execution of two subprocesses (abstract transitions in a model)

$A_r = A_1 \cup \cdots \cup A_n$, and each subset A_i is assigned an abstract event name a_i from a given set A of abstract names.

We would like to construct a model with transitions labeled by names from A, for which there exists a refinement via some substitution $[a_1 \to N_r^1, a_2 \to N_r^2, \ldots a_n \to N_r^n]$ respecting subset abstract names (i.e. if a transition t labeled by a_i is substituted by a net N_r^i, then transitions in N_r^i are labeled by names from A_i), such that the given log L conforms the refined model.

We suppose also, that the model we discover is acyclic. This assumption is due to the fact that several occurrences of low-level events belonging to the same subset A_i may be caused both by cyclic repetition of the abstract event a_i, and by executing a_i concurrently with some other abstract event via interleaving (cf. Fig. 6). So, in this paper we consider only acyclic case.

Discovery of a high-level model from a given event log will be done in two stages. First we construct a high-level transition system corresponding to the given low-level log, and then we use the well-known method based on theory of regions to discover a WF-net model from the transition system.

Now we explain our approach, the detailed algorithm is presented in the next section.

To construct a high-level transition system we first replace each activity in the given low-level log by the corresponding abstract activity. After this replacement we obtain the log, containing only high-level activities. However obtained traces may contain "stuttering" subsequences, when the same action occurs several times sequentially. This happens when several low-level activities corresponding to the same abstract activity go one by one in a trace. Such a "stuttering" subsequence should by replaced by one abstract activity.

However, removing stuttering is not enough to obtain a correct high-level log, where each abstract activity occurrence in a trace corresponds to one abstract event firing. If there are concurrent abstract actions in a system, representing subprocesses run in parallel, then low-level activities of these subprocesses may be interleaved in a trace. Then after replacing low-level activities by abstract ones and after removing stuttering we may still have a trace with several occurrences of the same abstract activity, which

actually correspond to one abstract event. An example of this is shown in Figure 6, where parallel high-level transitions $a1$ and $a2$ contain low-level transitions $t1, t2$ and $t3, t4$ respectively. Suppose the log contains the trace fragment: $< ..., t1, t3, t2, t4, ... >$. After substituting the corresponding abstract activities we get $< ..., a1, a2, a1, a2, ... >$. There is no stuttering, but two occurrences of $a1$ in the trace in fact correspond to one firing. The same is true for $a2$.

Thus, interleaving generates several occurrences of an abstract activity in a trace, but a repetition of activities may be caused also by cyclic behavior. To separate the concerns we suppose in this paper, that our system does not contain cycles. This can be easily checked: in acyclic system there are no repeated events in low-level logs.

So, we would like to construct an abstract log — a high-level version of a given low-level log with the following properties:

1. each abstract activity occurs not more than one time in each trace;
2. replacing of low-level activities by abstract ones should preserve interleaving of concurrent activities.

For that we propose the following solution. Each trace σ with k occurrences of the same abstract activity e is split into k clones of σ, where each clone is obtained by deleting all except one occurrences of e in σ. This is done for all repeated activities.

Let us illustrate this procedure by a small example. Suppose that after replacing low-level activities by abstract activities we have got two traces:
$\sigma_1 =< e0, e1, e3, e1, e3, e1, e4, e1, e7 >$,
$\sigma_2 =< e0, e1, e3, e1, e3, e1, e4, e6 >$.

Then by cloning these traces we get:
$\sigma_1^1 =< e0, e1, e3, e4, e7 >$,
$\sigma_1^2 =< e0, e3, e1, e4, e7 >$,
$\sigma_1^3 =< e0, e3, e4, e1, e7 >$,
$\sigma_2^1 =< e0, e1, e3, e4, e6 >$,
$\sigma_2^2 =< e0, e3, e1, e4, e6 >$.

Figure 7 shows the transition system obtained by convolution of the obtained traces. To synthesize a model we can just apply one of the known discovery algorithms to this transition system.

The precise description of the algorithm, based on this idea, is given in the next section.

6. Algorithm for Synthesis of Abstract Process Model from a Low-Level Event Log

Let A be a set of abstract activities and L_r be an event log (a finite multiset of traces) over a set of low-level activities from A_r, where traces do not contain repetitive activities. Let $\delta : A_r \to A$ be a function, which maps every low-level activity to some high-level activity from A. Start with the empty transition system TS.

Step 1. Convert L_r into an event log L over the set of activities A by replacing each activity $a \in L_r$ in each trace with the activity $\delta(a)$.

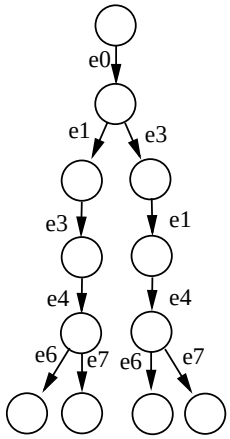


Fig 7. Transition system for traces σ_1 and σ_2

Step 2. Get rid of 'stuttering' in every $\sigma \in L$ by replacing in each trace each substring of consecutive entries of the same activity with one occurrence of this activity.

Step 3. For each trace $\sigma \in L$ do:

Step 3.1 Check whether there are more than one occurrences of the same activity in σ. If there are such occurrences, go to the Step 3.2, otherwise go to the Step 3.3.

Step 3.2 For each occurrence of repetitive activity e create a new trace by deleting all other occurrences of e in σ and go to Step 3.1.

Step 3.3 Add the new trace to transition system TS using the complete prefix sequence representation of a state.

Step 4 Apply an existing synthesis algorithm to the obtained transition system TS.

The correctness of the algorithm is justified by the following statements.

Lemma 1. *Let L_r be a low-level log without repetitive activities in its traces. If after Step 2 in our algorithm a trace $\sigma \in L$ contains two occurrences of activity a_1 and an occurrence of activity a_2 somewhere between two occurrence of a_1, then activities a_1 and a_2 are concurrent.*

Proof. Each occurrence of an abstract activity in a trace from L after Step 2 is obtained by replacing some low-level activity, corresponding to this abstract activity. So, occurrence of a_2 between two occurrences of a_1 means that at least a part of a_2 was executed, when execution of a_1 has started, but has not finished. Thus a_1 and a_2 were executed concurrently. □

Theorem 1. *Let N_{abs} be a WF-net with a set of activities A_{abs}, and let N_{ref} be a refinement of N_{abs} with a set of activities A_{ref} via substitutions $[a_1 \to N_r^1, a_2 \to N_r^2, \ldots, a_k \to M_r^k]$. Let also L_r be a log over the set of activities A_{ref}, and let L_r perfectly fit N_{ref}. Then each trace, added to the constructed transition system at Step 3.3, perfectly fits N_{abs}.*

Proof. Let $\sigma \in L$ be a trace added to transition system TS at Step 3.3. Then σ is obtained from some trace $\sigma_r \in L_r$ by replacing low-level activities with corresponding abstract activities and removing some repetitions of activities. Trace σ_r perfectly fits N_{ref}, and hence can be replayed with this model. Then σ be replayed with N_{abs} in parallel with replaying σ_r with N_{ref}. An activity a_i with the only occurrence at Step 3.1 of the algorithm will be replayed in parallel with a sequence of activities of subnet N_r^i in σ_r, and since A_{ref} is a refinement of N_{abs} via substitution $a_i \to N_r^i$, this replaying will be correct. If an activity a_j had multiple occurrences at this step, and the trace σ has retained one of this occurrences, it can be still replayed, since by lemma a_j is concurrent to its adjacent activities in σ, and concurrent activities can be replayed in any order. □

It can be also shown that the size of the constructed high-level transition system does not exceed the size of the low-level transition system, since the depth of the tree is shorter after abstraction. Note that if we want to keep real relative significance [15] of each arc in our model, we have to take into account that every trace which we generate at the Step 2.3 has an integer significance factor. This coefficient depends on the number of traces which are generated from one original trace from L_r (for k generated trace a significance factor equals to $1/k$ for each new trace).

7. Conclusion

Information system usually generate detailed event logs, which are not easy to work with. Detailed models discovered from these logs are often intricate and confusing. Abstract models are much more clear and more convenient for experts. So, the problem of discovering an abstract, high-level model from a low-level event log is important for simplification of the experts work on analysis and enhancement of information systems.

In this paper we provide a discovery technique for solving this problem, which is based on transforming a low-level event log into an abstract transition system and then applying one of already known methods for Petri net synthesis. In the further research we plan to evaluate this technique on artificial and real logs, using the fitness criteria presented earlier in [27].

References

[1] W. M. P. van der Aalst, *Process mining: discovery, conformance and enhancement of business processes*, Springer Verlag, 2011.

[2] B. F. van Dongen, A. K. A. de Medeiros, H. M. W. Verbeek, A. J. M. M. Weijters, W. M. P. van der Aalst, "The prom framework: A new era in process mining tool support", *International Conference on Application and Theory of Petri Nets*, Springer, 2005, 444–454.

[3] W. M. P. van der Aalst, "Verification of workflow nets", *18th International Conference on Application and Theory of Petri Nets, ICATPN'97*, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997, 407–426.

[4] S. Smirnov, H. A. Reijers, M. Weske, Th. Nugteren, "Business process model abstraction: a definition, catalog, and survey", *Distributed and Parallel Databases*, **30**:1 (2012), 63–99.

[5] A. Polyvyanyy, S. Smirnov, M. Weske, "Process model abstraction: A slider approach", *Enterprise Distributed Object Computing Conference, 2008* (EDOC'08. 12th International IEEE), IEEE, 2008, 325–331.

[6] K. Jensen, L. M. Kristensen, *Coloured Petri nets: modelling and validation of concurrent systems*, Springer Science & Business Media, 2009.

[7] H. J. Genrich, K. Lautenbach, "System modelling with high-level petri nets", *Theoretical computer science*, **13**:1 (1981), 109–135.

[8] W. M. P. van der Aalst, V. Rubin, B. F. van Dongen, E. Kindler, Ch. W. Günther, "Process mining: A two-step approach using transition systems and regions", *BPM Center Report BPM-06-30, BPMcenter. org*, **6**, 2006.

[9] A. J. M. M. Weijters, W. M. P. van der Aalst, A. K. A. De Medeiros, "Process mining with the heuristics miner-algorithm", *Technische Universiteit Eindhoven, Tech. Rep. WP*, **166**, 2006, 1–34.

[10] B. F van Dongen, N. Busi, G. Pinna, W. M. P. van der Aalst, "An iterative algorithm for applying the theory of regions in process mining" (Proceedings of the workshop on formal approaches to business processes and web services (FABPWS'07)), 2007.

[11] J. Carmona, J. Cortadella, M. Kishinevsky, "A Region-Based Algorithm for Discovering Petri Nets from Event Logs", *International Conference on Business Process Management*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, 358–373.

[12] W. M. P. van der Aalst, A. K. A. de Medeiros, A. J. M. M. Weijters, "Genetic process mining", *International Conference on Application and Theory of Petri Nets*, Springer, 2005, 48–69.

[13] A. J. M. M. Weijters, W. M. P. van der Aalst, "Rediscovering workflow models from event-based data using little thumb", *Integrated Computer-Aided Engineering*, **10**:2 (2003), 151–162.

[14] G. Greco, A. Guzzo, L. Pontieri, "Mining taxonomies of process models", *Data & Knowledge Engineering*, **67**:1 (2008), 74–102.

[15] Ch. W. Günther, W. M. P. van der Aalst, "Fuzzy mining–adaptive process simplification based on multi-perspective metrics", *International Conference on Business Process Management*, Springer, 2007, 328–343.

[16] F. Mannhardt, M. de Leoni, H. A. Reijers, W. M. P. van der Aalst, P. J. Toussaint, "From Low-Level Events to Activities – A Pattern-Based Approach", *Business Process Management: 14th International Conference, BPM 2016*, Springer International Publishing, Cham, 2016, 125–141.

[17] N. Tax, N. Sidorova, R. Haakma, W. M. P. van der Aalst, "Event abstraction for process mining using supervised learning techniques", *Proceedings of the SAI Conference on Intelligent Systems (IntelliSys)*, 2016, 161–170.

[18] J. Li, R. P. J. Ch. Bose, W. M. P. van der Aalst, "Mining context-dependent and interactive business process maps using execution patterns", *International Conference on Business Process Management*, Springer, 2010, 109–121.

[19] R. P. J. Ch. Bose, E. H. M. W. Verbeek, W. M. P. van der Aalst, "Discovering hierarchical process models using prom", *Forum at the Conference on Advanced Information Systems Engineering (CAiSE)*, Springer, 2011, 33–48.

[20] J. Cortadella, M. Kishinevsky, A. Kondratyev, L. Lavagno, A. Yakovlev, "Petrify: a tool for manipulating concurrent specifications and synthesis of asynchronous controllers", *IEICE Transactions on information and Systems*, **80**, 1997, 315–325.

[21] A. Kalenkova, I. Lomazova, "Discovery of cancellation regions within process mining techniques", *Fundamenta Informaticae*, **133**:2–3 (2014), 197–209.

[22] A. Kalenkova, I. Lomazova, W. M. P. van der Aalst, "Process Model Discovery: A Method Based on Transition System Decomposition", *International Conference on Application and Theory of Petri Nets*, Springer International Publishing, Cham, 2014, 71–90.

[23] R. P. J. Chandra Bose, W. M. P. van der Aalst, "Process diagnostics using trace alignment: opportunities, issues, and challenges", *Information Systems*, **37**:2 (2012), 117–141.

[24] Th. Baier, J. Mendling, "Bridging abstraction layers in process mining by automated matching of events and activities", *Business Process Management*, Springer, 2013, 17–32.

[25] J. Desel, W. Reisig, "The synthesis problem of petri nets", *Acta informatica*, **33**:4 (1996), 297–315.

[26] A. Rozinat, *Process mining: conformance and extension*, PhD thesis, Technische Universiteit Eindhoven, 2010.

[27] A. Begicheva, I. Lomazova, Does your event log fit the high-level process model? *Modeling and Analysis of Information Systems*, **22**:3 (2015), 392–403.

Бегичева А. К., Ломазова И. А., "Построение высокоуровневой модели процесса по журналу событий", *Моделирование и анализ информационных систем*, **24**:2 (2017), 125–140.

DOI: 10.18255/1818-1015-2017-2-125-140

Аннотация. Извлечение и анализ процессов (process mining) — это достаточно новая область компьютерных наук, изучающая синтез и анализ процессов на основе журналов событий. В работе рассматривается задача извлечения высокоуровневой модели по низкоуровневому журналу событий, т.е. задача автоматического синтеза модели процесса на основе информации, хранящейся в журналах событий информационной системы. События в высокоуровневой модели — это абстрактные события, которые могут быть детализированы в виде низкоуровневых подпроцессов, поведение которых представлено в журналах событий. Синтез моделей интенсивно изучается в рамках исследований по майнингу процессов, но в основном в литературе рассматриваются только логи и модели одного и того же уровня детализации. Здесь мы представляем алгоритм для извлечения высокоуровневых ациклических моделей процессов на основании журналов событий и заранее определенного разбиения низкоуровневых событий на подмножества, ассоциированные с абстрактными событиями в высокоуровневой модели.

Ключевые слова: сети Петри, высокоуровневые модели процессов, журналы событий, Process Mining, синтез моделей

Об авторах:
Бегичева Антонина Константиновна, стажер-исследователь,
Национальный исследовательский университет «Высшая школа экономики»,
Научно-учебная лаборатория ПОИС,
ул. Мясницкая, 20, г. Москва, 101000 Россия, e-mail: akbegicheva@edu.hse.ru

Ломазова Ирина Александровна, доктор физ.-мат. наук, профессор,
Национальный исследовательский университет «Высшая школа экономики»,
ул. Мясницкая, 20 г. Москва, 101000 Россия, e-mail: ilomazova@hse.ru

Благодарности:
Работа выполнена при поддержке Программы фундаментальных исследований Национального исследовательского университета «Высшая школа экономики» и Российского фонда фундаментальных исследований, проект 16-01-00546.

Моделирование и анализ информационных систем. Т. 24, № 2 (2017), с. 141–154
Modeling and Analysis of Information Systems. Vol. 24, No 2 (2017), pp. 141–154

DOI: 10.18255/1818-1015-2017-2-141-154

УДК 519.16 + 514.172.45

Полиэдральные характеристики задач о сбалансированном и несбалансированном двудольных подграфах

Бондаренко В. А.[1,2], **Николаев А. В.**[3], **Шовгенов Д. А.**[1]

получена 25 августа 2016

Аннотация. Исследуются полиэдральные характеристики трех задач о построении оптимальных полных двудольных подграфов двудольных графов. В первой задаче рассматриваются сбалансированные подграфы с одинаковым числом вершин в каждой доле и произвольными весами ребер. В двух других задачах речь идет о несбалансированных подграфах максимального и минимального веса с неотрицательными ребрами. Устанавливается, что все три задачи являются NP-трудными. В работе изучаются многогранники и конусные разбиения рассматриваемых задач, а также их графы. Для задачи о сбалансированном подграфе приводится условие смежности вершин в полиэдральном графе и графе соответствующего конусного разбиения. Плотность полиэдрального графа оценивается снизу сверхполиномиальной функцией. Для задач о несбалансированных подграфах строятся сверхполиномиальные нижние оценки плотности графов неотрицательных конусных разбиений. Полученные результаты характеризуют временную трудоемкость задач в широком классе алгоритмов, использующих линейные сравнения.

Ключевые слова: полный двудольный граф, полиэдральный граф, конусное разбиение, плотность графа, NP-трудная задача

Для цитирования: Бондаренко В. А., Николаев А. В., Шовгенов Д. А., "Полиэдральные характеристики задач о сбалансированном и несбалансированном двудольных подграфах", *Моделирование и анализ информационных систем*, **24**:2 (2017), 141–154.

Об авторах:
Бондаренко Владимир Александрович, orcid.org/0000-0002-5976-3446, д-р физ.-мат. наук, профессор,
Ярославский государственный университет им. П.Г. Демидова,
ул. Советская, 14, г. Ярославль, 150003, Россия, e-mail: bond@bond.edu.yar.ru

Николаев Андрей Валерьевич, orcid.org/0000-0003-4705-2409, канд. физ.-мат. наук,
Ярославский государственный университет им. П.Г. Демидова,
ул. Советская, 14, г. Ярославль, 150003, Россия, e-mail: andrei.v.nikolaev@gmail.com

Шовгенов Джамболет Азаматович, orcid.org/0000-0003-2022-4514, аспирант,
Ярославский государственный университет им. П.Г. Демидова,
ул. Советская, 14, г. Ярославль, 150003, Россия, e-mail: djsh92@mail.ru

Благодарности:
[1] При частичной поддержке гранта РФФИ № 14-01-00333.
[2] При частичной поддержке инициативной НИР ВИП-004 АААА-А16-116070610022-6.
[3] При поддержке гранта Президента Российской Федерации МК-5400.2015.13.

Введение

Рассматривается известная комбинаторная задача о существовании сбалансированного полного двудольного подграфа.

Сбалансированный полный двудольный подграф (balanced complete bipartite subgraph, BCBS). Заданы двудольный граф $G = (U, V, E)$ и положительное целое $k \leq |U| + |V|$. Существуют ли такие подмножества $U_x \subseteq U$ и $V_x \subseteq V$, что $|U_x| = |V_x| = k$ и $\{u, v\} \in E$ для любой пары вершин $u \in U_x$, $v \in V_x$.

К этой задаче сводится задача о клике, и она является NP-полной [12, 14]. Оптимизационная версия задачи, в которой требуется найти сбалансированный полный двудольный подграф с максимальным числом вершин, крайне трудна с точки зрения аппроксимации, для нее не известно ни одного хорошего приближенного алгоритма [11]. Значительное число работ посвящено изучению полиномиально разрешимых частных случаев задачи [7, 17]. Заметим, что полный двудольный граф также для сокращения называют бикликой [10].

Интересным является тот факт, что близкая задача о несбалансированном полном двудольном подграфе (замена условия $|U_x| = |V_x| = k$ на $|U_x| + |V_x| = k$) полиномиально разрешима. В основе алгоритма лежит известная теорема Кёнига [10].

Теорема 1. *(Кёниг) Для двудольного графа число рёбер в максимальном паросочетании совпадает с числом вершин в минимальном вершинном покрытии.*

Максимальное паросочетание в двудольном графе может быть найдено за время $O(|E|\sqrt{|U| + |V|})$ по алгоритму Хопкрофта–Карпа [16]. Доказательство теоремы Кёнига конструктивное и позволяет построить вершинное покрытие. Все вершины, не попавшие в минимальное вершинное покрытие, образуют максимальное независимое множество. Если рассмотреть дополнение двудольного графа, то независимое множество превратится в полный двудольный подграф.

Задачи, связанные с построением полных двудольных подграфов, часто возникают в различных прикладных областях. В частности, в вычислительной биологии для бикластеризации генов строится двудольный граф, в одну долю которого помещаются гены, а во вторую их свойства, и требуется построить наибольшую биклику [9]. Задача BCBS возникает также при проектировании интегральных схем (VLSI) для минимизации размера программируемых логических матриц (PLA) [18].

Рассмотрим три оптимизационных версии задачи о биклике во взвешенном двудольном графе.

Взвешенный сбалансированный полный двудольный подграф (weighted balanced complete bipartite subgraph, WBCBS). Заданы: полный двудольный граф $G = (U, V, E)$, $|U| = |V| = n$, функция весов $C : E \to \mathbb{R}$ и положительное целое $k \leq n$. Найти: сбалансированный полный двудольный подграф $G_x = (U_x, V_x, E_x)$ с максимальным (минимальным) суммарным весом ребер, при условии, что $|U_x| = |V_x| = k$.

Максимальный взвешенный полный двудольный подграф (maximum weighted complete bipartite subgraph, maxWCBS). Заданы: полный двудольный граф $G = (U, V, E)$, $|U| = |V| = n$, функция весов $C : E \to \mathbb{R}_+$ и положительное целое $k \leq 2n$. Найти: полный двудольный подграф $G_x = (U_x, V_x, E_x)$ с максимальным суммарным весом ребер, при условии, что $|U_x| + |V_x| = k$.

Минимальный взвешенный полный двудольный подграф (minimum weighted complete bipartite subgraph, minWCBS). Заданы: полный двудольный граф $G = (U, V, E)$, $|U| = |V| = n$, функция весов $C : E \to \mathbb{R}_+$ и положительное целое $k \leq 2n$. Найти: полный двудольный подграф $G_x = (U_x, V_x, E_x)$ с минимальным суммарным весом ребер, при условии, что $|U_x| + |V_x| = k$.

Отметим, что для задачи о сбалансированной биклике вопрос о том, рассматривается ли задача на минимум или на максимум, не является принципиальным. В обоих случаях решением будет k-сбалансированный полный двудольный подграф. Далее ограничимся рассмотрением только варианта на максимум. Задача на минимум может быть получена из него инвертированием знака у весов ребер.

В то же время для несбалансированного случая задачи на минимум и максимум являются принципиально разными. Действительно, если мы рассмотрим граф G с одинаковыми положительными весами ребер (который можно достроить до полного двудольного графа нулевыми ребрами для задачи на максимум и ребрами веса ∞ для задачи на минимум), то максимум достигается на сбалансированной или почти сбалансированной биклике с наибольшим числом ребер, а минимум на наиболее несбалансированной из возможных с наименьшим числом ребер. Для того, чтобы учесть эту особенность, далее рассматриваются постановки задач с неотрицательными весами ребер.

1. Конусные разбиения

Объектом исследования является конструкция конусного разбиения. Пусть X – некоторое конечное множество точек $\mathbb{R}^d$. Рассматривается задача максимизации линейной целевой функции на множестве X:

$$\langle c, x\rangle \to \max,\ x \in X.$$

Обозначим

$$K(x) = \{c \in \mathbb{R}^d :\ \langle c, x\rangle \geq \langle c, y\rangle\ ,\ \forall y \in X\}. \tag{1}$$

$K(x)$ является множеством решений конечной системы линейных однородных неравенств и представляет собой выпуклый многогранный конус. Учитывая, что

$$\bigcup_{x \in X} K(x) = \mathbb{R}^d,$$

совокупность всех конусов $K(x)$ называется *конусным разбиением* пространства $\mathbb{R}^d$ по множеству X. Конусное разбиение является аналогом диаграммы Вороного, в точности совпадая с ней, если евклидовы нормы всех точек множества X равны между собой.

Рассмотрим граф конусного разбиения. Его вершинами служат конусы разбиения, а два конуса $K(x)$ и $K(y)$ являются смежными тогда и только тогда, когда они имеют общую гиперграни:

$$\dim(K(x) \cap K(y)) = d - 1.$$

Обозначим через $\omega(X)$ плотность, или кликовое число, т.е. число вершин в наибольшей клике, графа конусного разбиения K пространства $\mathbb{R}^d$ по множеству X.

Известно [2], что трудоемкость алгоритмов прямого типа, использующих только линейные сравнения, по поиску максимума (или минимума, если поменять знак неравенства в определении конуса) линейной целевой функции $\langle c, x\rangle$ на множестве X, или, что то же самое, нахождению конуса $K(x)$, которому принадлежит целевой вектор c, не может быть меньше значения $\omega(X) - 1$. Таким образом, $\omega(X)$ является некоторой условной характеристикой сложности задач комбинаторной оптимизации в широком классе алгоритмов.

Определим через $M(X)$ выпуклую оболочку X: $M(X) = \mathrm{conv}(X)$. Выпуклой оболочкой конечного множества точек служит выпуклый многогранник, который называется многогранником задачи. Отметим, что для конусного разбиения над всем пространством $\mathbb{R}^d$ (1) имеет место следующее утверждение [2].

Лемма 1. *Две вершины x и y многогранника $M(X)$ смежны тогда и только тогда, когда конусы $K(x)$ и $K(y)$ имеют общую гиперграань.*

Таким образом, для конусного разбиения пространства $\mathbb{R}^d$ по множеству X граф совпадает с полиэдральным графом многогранника $M(X)$, множеством вершин которого служит множество геометрических вершин (в данном случае это X), а множеством ребер — совокупность геометрических ребер, т.е. множество одномерных граней.

Аналогично строятся конусные разбиения положительного ортанта $\mathbb{R}^d_+$ для задач на максимум и минимум

$$K^+_{\max}(x) = \{c \in \mathbb{R}^d_+ : \ \langle c, x\rangle \geq \langle c, y\rangle \,, \ \forall y \in X\}, \tag{2}$$

$$K^+_{\min}(x) = \{c \in \mathbb{R}^d_+ : \ \langle c, x\rangle \leq \langle c, y\rangle \,, \ \forall y \in X\}. \tag{3}$$

Эта конструкция, в свою очередь, является двойственной к полиэдру задачи, который определяется как доминанта выпуклой оболочки множества X:

$$\mathrm{dmt}(X) = \mathrm{conv}(V) + \mathbb{R}^d_+,$$

и применяется при анализе задач с неотрицательными исходными данными [13]. В нашем случае это неотрицательные веса ребер.

Изучению полиэдральных графов многогранников, графов конусных разбиений и их взаимосвязи со сложностью задач комбинаторной оптимизации посвящено большое число работ. В частности, были получены результаты для многогранников задачи коммивояжера [1] и задач об остовном дереве при дополнительных ограничениях [4], а также для неотрицательных конусных разбиений для задач о кратчайшем и самом длинном пути [5], задачи о разрезе [3, 8] и многих других [2].

2. Сбалансированный полный двудольный подграф

Теорема 2. *Задача WBCBS является NP-трудной.*

Доказательство. Ограничимся рассмотрением задачи на максимум. Рассуждения для случая на минимум проводятся полностью аналогично.

Рассмотрим двудольный граф $G = (U, V, E)$, $|U| = |V| = n$. Построим взвешенный полный двудольный граф $G^* = (U, V, E^*)$ с функцией весов вида:

$$c(i,j) = \begin{cases} 1, \text{ если } (i,j) \in E, \\ 0, \text{ в противном случае.} \end{cases}$$

В графе G^* найдется такая биклика x, что $|U_x| = |V_x| = k$, с суммарным весом ребер k^2 (это максимально возможный вес k-сбалансированного полного двудольного подграфа) тогда и только тогда, когда x является бикликой в графе G. NP-полная задача BCBS полиномиально сводится к задаче WBCBS, соответственно последняя является NP-трудной. □

Замечание: в статье для удобства используются веса ребер вида -1, 0, 1 и $+\infty$. Отметим, что в силу конечности числа ребер в полном графе последние можно заменить на такие целые положительные веса $\{a_1, a_2, a_3, a_4\}$, что для любого i вес a_i строго больше суммарного веса всех ребер a_{i-1}.

Каждому допустимому решению x задачи WBCBS, то есть каждому k-сбалансированному подграфу графа G, сопоставим характеристический вектор из пространства $\mathbb{R}^{n^2}$ по следующему правилу:

$$x_{i,j} = \begin{cases} 1, \text{ если } i \in U(x), j \in V(x), \\ 0, \text{ в противном случае.} \end{cases}$$

Обозначим через $X_{n,k}$ множество характеристических векторов всех допустимых решений и рассмотрим многогранник задачи о взвешенном сбалансированном полном двудольном подграфе $WBCBS(n,k) = \mathrm{conv}(X_{n,k})$, а также конусное разбиение $K_{n,k}$ пространства $\mathbb{R}^{n^2}$ по множеству $X_{n,k}$. Через $c \in \mathbb{R}^{n^2}$ определим вектор с весами ребер графа G. Тогда суммарный вес ребер подграфа G_x равен значению целевой функции $\langle c, x \rangle$.

Лемма 2. *Две вершины x и y многогранника $WBCBS(n,k)$ смежны тогда и только тогда, когда соответствующие двудольные подграфы не имеют общих долей:*

$$U(x) \neq U(y) \text{ и } V(x) \neq V(y),$$

либо доли в одной части совпадают, а в другой отличаются не более чем одной вершиной:

$$\left[\begin{array}{l} U(x) = U(y), |V(x) \backslash V(y)| = 1, \\ V(x) = V(y), |U(x) \backslash U(y)| = 1. \end{array} \right.$$

Доказательство. В силу Леммы 1 смежность вершин многогранника $WBCBS(n,k)$ равносильна смежности конусов в разбиении $K_{n,k}$. Проведем доказательство с точки зрения конусного разбиения.

Пусть $x, y \in X_{n,k}$. Смежность конусов $K_{n,k}(x)$ и $K_{n,k}(y)$ означает, что найдется такой вектор $c \in \mathbb{R}^{n^2}$, что в конусном разбиении $K_{n,k}$ вектор c принадлежит исключительно конусам $K_{n,k}(x)$ и $K_{n,k}(y)$ (конусы имеют общую гиперграни)

$$\exists c \in \mathbb{R}^{n^2}, \forall z \in X_{n,k} \backslash \{x, y\} : \langle c, x \rangle = \langle c, y \rangle > \langle c, z \rangle . \quad (4)$$

Пусть подграфы x и y не имеют общих долей. Построим вектор весов ребер c по следующему правилу (Рис. 1):

$$c_{i,j} = \begin{cases} 1, \text{ если } i \in U(x), j \in V(x) \text{ или } i \in U(y), j \in V(y), \\ 0, \text{ в противном случае.} \end{cases} \tag{5}$$

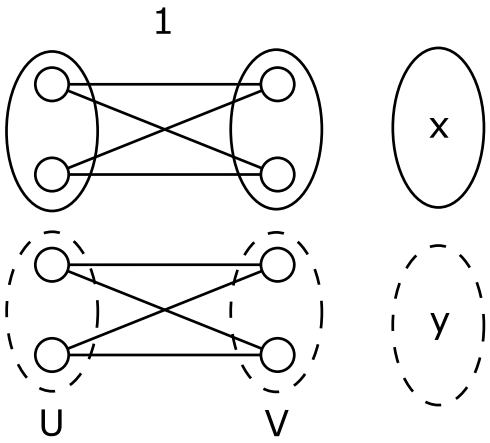


Рис. 1: Функция весов для сбалансированного подграфа без общих долей
Fig. 1. The weight function for a balanced subgraph without common parts

В таком случае получаем

$$\langle c, x \rangle = \langle c, y \rangle = k^2,$$

и это максимально возможный вес k-сбалансированной биклики в графе.

Рассмотрим произвольный подграф $z \in X_{n,k} \backslash \{x, y\}$.

- Если z включает хотя бы одну вершину в доле U, не принадлежащую $U(x) \cup U(y)$, то $\langle c, z \rangle < k^2$, так как все ребра инцидентные этой вершине имеют нулевой вес.
- Если z включает в доле U как вершины из $U(x) \backslash U(y)$, так и из $U(y) \backslash U(x)$, тогда в правой доле V лишь вершины $V(x) \cap V(y)$ имеют с ними ненулевые ребра одновременно. Но $|V(x) \cap V(y)| < k$, а значит, хотя бы одно ребро имеет нулевой вес, и $\langle c, z \rangle < k^2$.
- Если $U(z) = U(x)$ $(U(z) = U(y))$, то следует аналогично рассмотреть долю V и показать, что при $z \neq x$ $(z \neq y)$ получаем $\langle c, z \rangle < k^2$.

Таким образом, конусы $K_{n,k}(x)$ и $K_{n,k}(y)$ смежны по условию (4).

Пусть подграфы x и y совпадают в одной доле. Без ограничения общности будем считать, что $V(x) = V(y)$. Пусть $|U(x) \backslash U(y)| = 1$. Построим вектор весов c следующего вида (Рис. 2):

$$c_{i,j} = \begin{cases} 1, \text{ если } i \in U(x) \cap U(y), j \in V(x) = V(y), \\ 0, \text{ если } i \in U(x) \triangle U(y), j \in V(x) = V(y), \\ -1, \text{ в противном случае.} \end{cases}$$

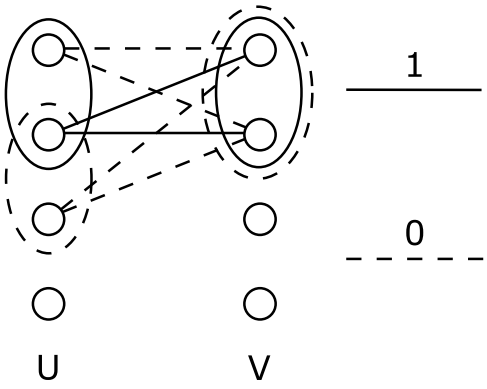


Рис. 2: Функция весов для сбалансированного подграфа с общей долей
Fig. 2. The weight function for a balanced subgraph with a common part

По построению получаем

$$\langle c, x\rangle = \langle c, y\rangle = k(k-1).$$

Рассмотрим произвольный подграф $z \in X_{n,k}\backslash\{x, y\}$.

- Если z включает хотя бы одну вершину в доле U, не принадлежащую $U(x) \cup U(y)$, то все ребра, инцидентные этой вершине, имеют отрицательный вес, и

$$\langle c, z\rangle \leq k(k-1) - k < k(k-1).$$

- Если z включает обе вершины из $U(x)\triangle U(y)$, то инцидентные им ребра имеют нулевой или отрицательный вес, и

$$\langle c, z\rangle \leq k(k-2) < k(k-1).$$

Соответственно, конусы $K_{n,k}(x)$ и $K_{n,k}(y)$ являются смежными.

Остается рассмотреть случай $|U(x)\backslash U(y)| \geq 2$. Обозначим через

$$a = |U(x) \cap U(y)| \leq k-2.$$

Тогда симметрическая разность $U(x)\triangle U(y)$ содержит $2(k-a)$ вершин из которых можно выбрать $k-a$ вершин

$$\binom{2(k-a)}{k-a} \geq 6$$

различными способами. Следовательно, найдутся такие подграфы $z, t \in X_{n,k}\backslash\{x, y\}$, что

$$\begin{gathered} V(z) = V(t) = V(x) = V(y), \\ U(z) \cap U(t) = U(x) \cap U(y), \\ U(z) \cup U(t) = U(x) \cup U(y). \end{gathered}$$

Таким образом,

$$\langle c, x\rangle + \langle c, y\rangle = \langle c, x \cup y\rangle + \langle c, x \cap y\rangle = \langle c, z \cup t\rangle + \langle c, z \cap t\rangle = \langle c, z\rangle + \langle c, t\rangle,$$

и хотя бы один из подграфов z или t имеет вес не меньший, чем x и y. Конусы $K_{n,k}(x)$ и $K_{n,k}(y)$ не являются смежными. □

Теорема 3. *Плотность полиэдрального графа многогранника $WBCBS(n,k)$ задачи о взвешенном сбалансированном полном двудольном подграфе сверхполиномиальна по параметрам n и k:*

$$\omega(WBCBS(n,k)) \geq \binom{n}{k} = \Omega\left(\left(\frac{n}{k}\right)^k\right).$$

Доказательство. Рассмотрим подмножество k-сбалансированных двудольных подграфов $Y_{n,k} \subset X_{n,k}$ следующего вида: занумеруем все вершины в каждой доле числами от 1 до n, и будем рассматривать лишь подграфы с одинаковыми номерами выбранных вершин в левой и правой долях. Любые два подграфа $x, y \in Y_{n,k}$ не имеют общих долей, следовательно, по Лемме 2 соответствующие вершины многогранника $WBCBS(n,k)$ являются смежными. Соответственно $Y_{n,k}$ образует клику в полиэдральном графе многогранника задачи

$$|Y_{n,k}| = \binom{n}{k}.$$

Асимптотическая нижняя оценка является стандартной для числа сочетаний. □

3. Максимальный полный двудольный подграф

Теперь обратимся к несбалансированному случаю.

Теорема 4. *Задача $maxWCBS$ является NP-трудной.*

Доказательство. Воспользуемся рассуждением для Теоремы 2. Рассмотрим двудольный граф G, всем ребрам назначим вес 1 и дополним его до полного взвешенного графа G^* ребрами нулевого веса. В графе G^* найдется биклика x веса k^2 тогда и только тогда, когда x является k-сбалансированной бикликой в графе G. Задача BCBS полиномиально сводится к задаче maxWCBS. □

По аналогии со сбалансированным случаем каждому допустимому решению задачи maxWCBS сопоставим характеристический вектор $x \in \mathbb{R}_+^{n^2}$. Обозначим через $X^u_{n,k}$ множество характеристических векторов всех допустимых решений и рассмотрим неотрицательное конусное разбиение $K^{max}_{n,k}$ положительного ортанта $\mathbb{R}_+^{n^2}$ по множеству $X^u_{n,k}$ для задачи на максимум (2).

Теорема 5. *Плотность графа конусного разбиения $K^{max}_{n,k}$ задачи о взвешенном полном двудольном подграфе сверхполиномиальна по параметрам n и k:*

$$\omega(K^{max}_{n,k}) \geq \binom{n}{s} = \Omega\left(\left(\frac{n}{s}\right)^s\right), \text{ для } k = 2s,$$

$$\omega(K^{max}_{n,k}) \geq \binom{n-1}{s} = \Omega\left(\left(\frac{n-1}{s}\right)^s\right), \text{ для } k = 2s+1.$$

Доказательство. Для четного k можно воспользоваться конструкцией из доказательства Теоремы 3 с одинаковым выбором вершин в левой и правой долях. Неотрицательные конусы s-сбалансированных биклик без одинаковых долей будут попарно смежны по условию Леммы 2. Отметим, что при доказательстве этой части критерия смежности использовались неотрицательные 0/1-веса (Рис. 1).

Вариант с нечетным k сводится к четному. Зафиксируем в левой и правой долях по одной вершине. Например, вершины с номером n. Рассмотрим множество $Y^{max}_{n,k}$ двудольных подграфов следующего вида:

$$|U_x| = s, U_x \subseteq \{1, 2, 3, \dots n-1\},$$
$$|V(x)| = s+1, \{n\} \in V(x),$$

содержащих вершину с номером n в правой доле.

Для любых двух подграфов x и y из $Y^{max}_{n,k}$ можно построить вектор весов c по правилу (5). Подграфы x и y имеют максимально возможный вес для биклик с $2s+1$ вершинами:

$$\langle c, x\rangle = \langle c, y\rangle = s(s+1).$$

Любой двудольный подграф z, включающий вершины не из $U(x) \cup U(y)$, или одновременно вершины из $U(x)\backslash U(y)$ и $U(y)\backslash U(x)$, будет иметь меньший вес. Следовательно, по условию (4), конусы $K^{max}_{n,k}(x)$ и $K^{max}_{n,k}(y)$ смежны, а $Y^{max}_{n,k}$ образует клику в графе конусного разбиения. Учитывая исключение вершины с номером n, случай нечетного k сводится к четному в графе с $n-1$ вершиной в каждой доле. □

Заметим, что все результаты и доказательства для задачи о максимальном полном двудольном подграфе повторяют аналогичные для сбалансированного случая. Это ожидаемо, учитывая, что для максимизации числа ребер (весов ребер) искомый подграф должен быть максимально близок к сбалансированному. Для задачи на минимум ситуация будет несколько иной.

4. Минимальный полный двудольный подграф

Теорема 6. *Задача minWCBS является NP-трудной.*

Доказательство. Рассмотрим двудольный граф G с ребрами двух возможных весов: 1 и n^2. Будем считать, что вместо выбора вершин в подграф мы исключаем вершины и все инцидентные им ребра из полного графа. Тогда задача minWCBS о поиске двудольного подграфа на k вершинах минимального веса эквивалентна задаче о поиске подмножества из $2n-k$ вершин с максимальным числом инцидентных ребер веса n^2 для исключения. Заметим, что суммарный вклад всех ребер веса 1 меньше вклада одного ребра веса n^2. Построим невзвешенный двудольный граф G^*, удалив из G все ребра единичного веса. Рассматриваемая задача является задачей о максимальном q-вершинном покрытии в графе G^* для $q = 2n-k$.

Максимальное q-вершинное покрытие (maximum q-vertex cover). Задан граф $G = (V, E)$ и положительное целое $q < |V|$. Найти подмножество вершин $U \subset V$ с наибольшим числом инцидентных ребер, такое что $|U| = q$.

Задача о максимальном q-вершинном покрытии является NP-трудной даже для двудольных графов [6, 15]. □

Рассмотрим конусное разбиение $K^{min}_{n,k}$ пространства $\mathbb{R}^{n^2}_+$ по множеству $X^u_{n,k}$ характеристических векторов допустимых решений minWCBS для задачи на минимум (3). Обозначим:

$$m = \left\lfloor \frac{n}{2} \right\rfloor.$$

Теорема 7. *Пусть*

$$k = 3s \ u \ \frac{9}{4}m < k < 3m, \tag{6}$$

тогда плотность графа конусного разбиения $K^{min}_{n,k}$ задачи о минимальном взвешенном полном двудольном подграфе сверхполиномиальна по параметрам n и k:

$$\omega(K^{min}_{n,k}) \geq \binom{m}{s} = \Omega\left(\left\lfloor \frac{3n}{2k} \right\rfloor^{\frac{k}{3}}\right).$$

Доказательство. Без ограничения общности будем считать, что $n = 2m$. В противном случае можно выбрать по вершине в каждой доле и исключить их из дальнейших построений.

Разобьем множества вершин в каждой доле на два равных непересекающихся подмножества $U = U_1 \cup U_2$, $V = V_1 \cup V_2$. Внутри каждого множества занумеруем вершины числами от 1 до m.

Рассмотрим множество $Y^{min}_{n,k}$ подграфов следующего вида: для любого подмножества

$$\{x_1, x_2, \ldots, x_s\} \subset \{1, 2 \ldots m\}$$

мы строим двудольный подграф x, включающий вершины с номерами $\{x_1, \ldots, x_s\}$ из множеств U_1, V_1 и V_2.

Покажем, что конусы решений из множества $Y^{min}_{n,k}$ являются попарно смежными. Рассмотрим два произвольных подграфа x, y из $Y^{min}_{n,k}$. Построим функцию весов ребер c следующего вида (Рис. 3):

$$c_{i,j} = \begin{cases} 0, \text{ если } i \in U(x)\backslash U(y), j \in V(x), \\ \quad \text{или } i \in U(y)\backslash U(x), j \in V(y), \\ \quad \text{или } i \in U(x) \cup U(y), j \in V(x) \cap V(y), \\ 1, \text{ если } i \in U(x) \cap U(y), j \in V(x) \triangle V(y), \\ \infty, \text{ в противном случае.} \end{cases}$$

Пусть $|U(x) \cap U(y)| = a$, тогда

$$\langle c, x \rangle = \langle c, y \rangle = 2a(s - a). \tag{7}$$

Отметим, что по построению множества $Y^{min}_{n,k}$:

$$2s - m \leq a < s. \tag{8}$$

Рассмотрим произвольный двудольный подграф z с числом вершин $k = 3s$.

- Если z включает хотя бы одну вершину в доле U, не принадлежащую $U(x) \cup U(y)$, то $\langle c, z \rangle = \infty$, так как все инцидентные этой вершине ребра имеют бесконечный вес.

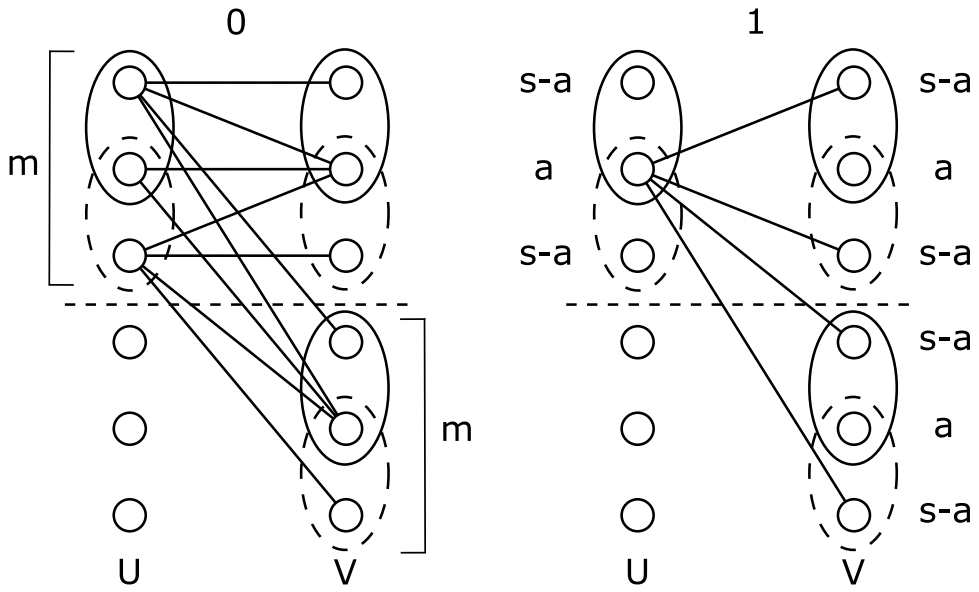


Рис. 3: Функция весов для минимального несбалансированного подграфа
Fig. 3. The weight function for a minimum unbalanced subgraph

- Если z включает в доле U как вершины из $U(x)\backslash U(y)$, так и из $U(y)\backslash U(x)$, то в правой доле V лишь вершины $V(x) \cap V(y)$ имеют с ними ребра конечного веса одновременно:

$$|V(x) \cap V(y)| = 2a,$$
$$|U(x) \cup U(y)| = 2s - a,$$
$$(2s - a) + 2a = 2s + a < 3s = k.$$

  Для двудольного подграфа z конечного веса в данном случае не хватает вершин.

- Если $U(z) \subseteq U(x)$, и z включает хотя бы одну вершину из $U(x)\backslash U(y)$, то в правой доле только вершины из $V(x)$ имеют с выбранной вершиной ребра конечного веса. В данном случае если $\langle c, z\rangle < \infty$, то $z = x$. Случай с вершиной из $U(y)\backslash U(x)$ рассматривается аналогично.

- Если $U(z) \subseteq U(x) \cap U(y)$, тогда

$$\langle c, z\rangle = b(3s - b - 2a), \tag{9}$$

  где $b = |U(z)|$. По построению величина b удовлетворяет следующим ограничениям:

$$3s - 2m \leq b \leq a. \tag{10}$$

  Предположим, что вес подграфа z (9) не превосходит весов x и y (7), тогда, с учетом неравенств (6, 8, 10), выполняется следующая система:

$$\begin{cases} b(3s-b-2a) \le 2a(s-a), \\ \frac{3}{4}m < s < m, \\ 2s-m \le a < s, \\ 3s-2m \le b \le a, \\ m, s, a, b > 0, \end{cases}$$

которая не имеет решений ни при каких значениях параметров m, s, a, b. Таким образом, вес любого подграфа z строго больше весов x и y. По условию (4) конусы $K^{min}_{n,k}(x)$ и $K^{min}_{n,k}(y)$ смежны, а $Y^{min}_{n,k}$ образует клику в графе конусного разбиения. По построению, получаем

$$|Y^{min}_{n,k}| = \binom{m}{s} = \Omega\left(\left\lfloor \frac{3n}{2k} \right\rfloor^{\frac{k}{3}}\right).$$

□

5. Заключение

Задачи о построении оптимальных двудольных подграфов исследовались многими авторами и имеют множество практических применений. Для трех рассматриваемых задач о сбалансированном подграфе с произвольными весами и несбалансированных подграфах минимального и максимального веса с неотрицательными весами установлены NP-полнота задач и сверхполиномиальные плотности графов многогранников и конусных разбиений. Во всех трех случаях полиэдральные характеристики коррелируют со сложностью задачи.

Отметим, что задача о минимальном полном двудольном подграфе является NP-трудной, при том, что близкая задача о существовании несбалансированного двудольного подграфа на k вершинах полиномиально разрешима. Возможно, именно в связи с этим задача minWCBS оказалась наиболее трудной для анализа, а вопрос о сложности двойственной задачи о максимальном q-вершинном покрытии в двудольном графе был открыт в течение многих лет [6].

Список литературы / References

[1] Бондаренко В. А., “Неполиномиальная нижняя оценка сложности задачи коммивояжера в одном классе алгоритмов”, *Автоматика и телемеханика*, **9** (1983), 45–50; [Bondarenko V. A., “Nonpolynomial lowerbound of the traveling salesman problem complexety in one class of algorithms”, *Automation and remote control*, **44**:9 (1983), 1137–1142.]

[2] Бондаренко В. А., Максименко А. Н., *Геометрические конструкции и сложность в комбинаторной оптимизации*, ЛКИ, М., 2008, 184 с.; [Bondarenko V. A., Maksimenko A. N., *Geometricheskie konstruktsii i slozhnost v kombinatornoy optimizatsii*, LKI, Moscow, 2008, (in Russian).]

[3] Бондаренко В. А., Николаев А. В., “Комбинаторно-геометрические свойства задачи о разрезе”, *Доклады Академии наук*, **452**:2 (2013), 127–129; [Bondarenko V. A., Nikolaev A. V., “Combinatorial and Geometric Properties of the Max-Cut and Min-Cut Problems”, *Doklady Mathematics*, **88**:2 (2013), 516–517.]

[4] Бондаренко В. А., Николаев А. В., Шовгенов Д. А., "Полиэдральные графы задач об остовных деревьях при дополнительных ограничениях", *Моделирование и анализ информационных систем*, **22**:4 (2015), 453–463; [Bondarenko V. A., Nikolaev A. V., Shovgenov D. A., "1-Skeletons of the Spanning Tree Problems with Additional Constraints", *Modeling and Analysis of Information Systems*, **22**:4 (2015), 453–463, (in Russian).]

[5] Максименко А. Н., "Комбинаторные свойства многогранника задачи о кратчайшем пути", *Ж. вычисл. матем. и матем. физ.*, **44**:9 (2004), 1693–1696; [Maksimenko A. N., "Combinatorial properties of the polyhedron associated with the shortest path problem", *Comput. Math. Math. Phys.*, **88**:2 (2013), 1611–1614.]

[6] Apollonio N., Simeone B., "The maximum vertex coverage problem on bipartite graphs", *Discrete Applied Mathematics*, **165** (2014), 37–48.

[7] Arbib C., Mosca R., "Polynomial algorithms for special cases of the balanced complete bipartite subgraph problem", *Journal of Combinatorial Mathematics and Combinatorial Computing*, **39** (1999), 3–22.

[8] Bondarenko V., Nikolaev A., "On Graphs of the Cone Decompositions for the Min-Cut and Max-Cut Problems", *International Journal of Mathematics and Mathematical Sciences*, **2016** (2016), 6 pages, Article ID 7863650.

[9] Cheng Y., Church G. M., "Biclustering of expression data", Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology, 2000, 93–103.

[10] Diestel R., *Graph Theory*, Springer-Verlag Berlin Heidelberg, 2010, 410 pp.

[11] Feige U., Kogan S., *Hardness of approximation of the Balanced Complete Bipartite Subgraph problem. Tech. Rep. MCS04-04*, Dept. of Comp. Sci. and Appl. Math., The Weizmann Inst. of Science, 2004.

[12] Garey M. R., Johnson D. S., *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman & Co., New York, NY, USA, 1979, 340 pp.

[13] Grötschel M., Lovasz L., Schrijver A., *Geometric Algorithms and Combinatorial Optimization*, Springer–Verlag Berlin Heidelberg, 1993, 362 pp.

[14] Johnson D. S., "The NP-completeness column: An ongoing guide", *Journal of Algorithms*, **8**:3 (1987), 438–448.

[15] Joret G., Vetta A., "Reducing the rank of a matroid", *Discrete Mathematics & Theoretical Computer Science*, **17**:2 (2015), 143–156.

[16] Hopcroft J. E., Karp R. M., "An $n^{5/2}$ Algorithm for Maximum Matchings in Bipartite Graphs", *SIAM Journal on Computing*, **2**:4 (1973), 225–231.

[17] Mubayi D., Turàn G., "Finding bipartite subgraphs efficiently", *Information Processing Letters*, **110**:5 (2010), 174–177.

[18] Ravi S. S., Lloyd E. L., "The complexity of near-optimal programmable logic array folding", *SIAM Journal on Computing*, **17**:4 (1988), 696–710.

Bondarenko V. A.[1,2], **Nikolaev A. V.**[3], **Shovgenov D. A.**[1], "Polyhedral Characteristics of Balanced and Unbalanced Bipartite Subgraph Problems", *Modeling and Analysis of Information Systems*, **24**:2 (2017), 141–154.

DOI: 10.18255/1818-1015-2017-2-141-154

Abstract. We study the polyhedral properties of three problems of constructing an optimal biclique in a bipartite graph. In the first problem we consider a balanced biclique with the same number of vertices in both parts and arbitrary edge weights. In the other two problems it is required to find maximum or minimum unbalanced bicliques with a fixed number of vertices and non-negative edges. All three problems are established to be NP-hard. We study the polytopes and the cone decompositions of these problems and their 1-skeletons. We describe the adjacency criterion in the 1-skeleton of the

balanced biclique polytope. Clique number of 1-skeleton is estimated from below by a superpolynomial function. For both unbalanced biclique problems we establish the superpolynomial lower bounds on the clique numbers of the graphs of non-negative cone decompositions. These values characterize the time complexity in a broad class of algorithms based on linear comparisons.

Keywords: biclique, 1-skeleton, cone decomposition, clique number, NP-hard problem

About the authors:
Vladimir Bondarenko, orcid.org/0000-0002-5976-3446, doctor of science, professor,
P.G. Demidov Yaroslavl State University,
14 Sovetskaya str., Yaroslavl 150003, Russia, e-mail: bond@bond.edu.yar.ru,

Andrei Nikolaev, orcid.org/0000-0003-4705-2409, PhD,
P.G. Demidov Yaroslavl State University,
14 Sovetskaya str., Yaroslavl 150003, Russia, e-mail: andrei.v.nikolaev@gmail.com

Dzhambolet Shovgenov, orcid.org/0000-0003-2022-4514, graduate student,
P.G. Demidov Yaroslavl State University,
14 Sovetskaya str., Yaroslavl 150003, Russia, e-mail: djsh92@mail.ru

Acknowledgments:
[1] Partially supported by the Russian Foundation for Basic Research project 14-01-00333.
[2] Partially supported by the initiative R&D VIP-004 AAAA-A16-116070610022-6.
[3] Supported by the President's of Russian Federation grant MK-5400.2015.1.

Моделирование и анализ информационных систем. Т. 24, № 2 (2017), с. 155–167
Modeling and Analysis of Information Systems. Vol. 24, No 2 (2017), pp. 155–167

DOI: 10.18255/1818-1015-2017-2-155-167

УДК 004.652.4

Анализ типизированных зависимостей включения с неопределенными значениями

Зыкин В. С., Зыкин С. В.

получена 24 августа 2016

Аннотация. Неопределенные значения стали актуальной проблемой с момента создания реляционной модели данных. Влияние неопределенностей сказывается на всех видах зависимостей, используемых при проектировании и эксплуатации базы данных. В полной мере это относится и к зависимостям включения, которые являются теоретической основой ссылочной целостности на данные. Попытки решения указанной проблемы содержат неточности как в постановке задачи, так и в самом ее решении. К постановочным ошибкам можно отнести использование в определении нетипизированных зависимостей включения, что приводит к перестановкам атрибутов, хотя в технологиях баз данных атрибуты идентифицируются по имени, а не по их позиции. Кроме того, связывание зависимостью включения разнородных, пусть даже однотипных, атрибутов является признаком потерянной функциональной зависимости и приводит к взаимодействию нетривиальных зависимостей включения и функциональных зависимостей. Зависимости включения должны определять количественное соотнесение объектов друг с другом, а не значений атрибутов. Неточности в решении указанной проблемы содержатся в формулировках аксиом и доказательстве их свойств, в том числе полноты. В этой статье предлагается оригинальное решение этой проблемы только для типизированных зависимостей включения при наличии неопределенных значений: предложена система аксиом, доказана ее полнота и непротиворечивость. На основе правил вывода разработан алгоритм построения не избыточного множества типизированных зависимостей включения. Доказана корректность этого алгоритма.

Ключевые слова: база данных, зависимости включения, аксиоматика, неопределенные значения

Для цитирования: Зыкин В. С., Зыкин С. В., "Анализ типизированных зависимостей включения с неопределенными значениями", *Моделирование и анализ информационных систем*, **24**:2 (2017), 155–167.

Об авторах:
Зыкин Владимир Сергеевич, orcid.org/0000-0002-6492-2464, аспирант,
Омский государственный технический университет,
просп. Мира, 11, г. Омск, 644050 Россия, e-mail: vszykin@mail.ru

Зыкин Сергей Владимирович, orcid.org/0000-0002-0576-2149, д-р техн. наук, профессор,
Институт математики им. С.Л. Соболева СО РАН,
ул. Певцова, 13, г. Омск, 644043 Россия, e-mail: szykin@mail.ru

Введение

Целостность базы данных (database integrity) – соответствие имеющейся в базе данных (БД) информации логике соответствующей прикладной области. Ссылочные ограничения целостности на данные (referential integrity) являются одним из основных видов ограничений, которые позволяют сохранить структурную целостность

БД. В большинстве существующих систем управления базами данных (СУБД) поддерживается такой вид ограничений, и задаются эти ограничения в виде связей (relationship) на схеме БД.

Теоретической основой ссылочных ограничений являются зависимости включения и их взаимодействие с другими видами зависимостей. Проблеме исследования зависимостей включения до сих пор уделяется внимание со стороны исследователей, поскольку, с одной стороны, остаются нерешенными некоторые теоретические проблемы. С другой стороны, практика использования БД формулирует новые требования к ссылочным ограничениям целостности.

Для методического сопровождения последующих формальных построений рассмотрим два примера схем БД.

Пример 1. Пусть задано некоторое множество отношений – фрагмент схемы БД учебного заведения, где подчеркнуты ключевые атрибуты отношений:
R_1 = *Студенты* (№ студента, № группы, ФИО студента);
R_2 = *Список групп* (№ группы, Код группы, № специальности, № курса);
R_3 = *Предметы* (№ предмета, Предмет);
R_4 = *Экзамен* (№ студента, № группы, № предмета, Оценка);
R_5 = *Аттестация* (№ студента, № группы, № предмета, Вид аттестации, Балл).

В примере 1 на схеме должны быть установлены следующие ссылочные ограничения целостности: нельзя поставить оценку за экзамен либо балл по аттестации по предмету, которого нет в отношении R_3. Кроме того, нельзя поставить оценку за экзамен либо балл по аттестации студенту, информацию о котором не внесли в БД. С другой стороны, между отношениями R_4 и R_5 могла быть установлена связь по атрибутам (№ студента, № группы, № предмета), которая содержательно задает следующее ограничение целостности: аттестация студентов может осуществляться только по тем предметам, по которым проставлена оценка. Однако это противоречит прикладной области, и такого ограничения быть не может.

Пример 2. Рассмотрим фрагмент схемы БД кинотеатра:
R_1 = *Фильмы* (№ фильма, Наименование фильма, Жанр фильма);
R_2 = *Расписание сеансов* (Дата сеанса, Время сеанса, № зала, № фильма);
R_3 = *Билеты* (№ билета, Дата сеанса, Время сеанса, № зала, № ряда, № места).

В примере 2 на схеме должны быть установлены следующие ссылочные ограничения целостности: нельзя в расписании назначить сеанс, если неизвестен фильм. Однако возможно продать билет со свободной (неопределенной) датой посещения, и/или свободным временем, и/или неопределенным номером зала и т.д. Это означает, что определенным значениям в отношении R_2 могут быть поставлены в соответствие неопределенные значения (Null) в R_3. И если появится необходимость заменить неопределенные значения в R_3 на определенные, то выбор может быть сделан только из соответствующих определенных значений в R_2.

В данной работе предлагается исследование формальной теории для типизированных зависимостей включения при наличии неопределенных значений. Разработанная теория используется для построения не избыточного множества ссылочных ограничений целостности.

1. Обзор результатов

Формирование структуры отношений (таблиц) на схеме БД осуществляется с использованием функциональных зависимостей, многозначных зависимостей и зависимостей соединения [1, 2]. Однако этих зависимостей недостаточно для установления ссылочных ограничений целостности между сформированными отношениями. Для этого перечисленные зависимости дополняются зависимостями включения. Формальное определение зависимостей включения приведено в работе [3]: пусть $U = \{A_1, A_2, \ldots, A_n\}$ – множество атрибутов, определенных в БД, $[R_i]$ – множество атрибутов, на которых определено отношение R_i, $[R_i] \subseteq U$, $1 \leq i \leq k$, $\Re = (R_1, R_2, \ldots, R_k)$ – БД, $S = \{[R_1], [R_2], \ldots, [R_k]\}$ – схема БД.

Определение 1. *Пусть $[R_i]$ и $[R_j]$ – схемы отношений (не обязательно различные), $V \subseteq [R_i]$ и $W \subseteq [R_j]$, $|V| = |W|$, тогда соотношение $R_i[V] \subseteq R_j[W]$ называется зависимостью включения.*

В определении 1 $|V|$ – мощность множества V, $R_i[V] = \pi_V(R_i)$ – проекция отношения R_i по атрибутам V. Зависимость включения считается типизированной, если $V = W$, в противном случае – нетипизированной.

В этой же работе [3] представлена система аксиом зависимостей включения:

- **IND1**) (рефлексивность): $R_i[X] \subseteq R_i[X]$, если X – последовательность отдельных атрибутов R_i.
- **IND2**) (проецирование и перестановка): если $R_i[A_1, \ldots, A_m] \subseteq R_j[B_1, \ldots, B_m]$, тогда $R_i[A_{i_1}, \ldots, A_{i_q}] \subseteq R_j[B_{i_1}, \ldots, B_{i_q}]$ для каждой последовательности $i_1, \ldots, i_q$ различных целочисленных значений из множества $\{1, \ldots, m\}$.
- **IND3**) (транзитивность): если $R_i[X] \subseteq R_j[Y]$ и $R_j[Y] \subseteq R_l[Z]$, тогда выполнено $R_i[X] \subseteq R_l[Z]$.

Относительно системы **IND1**–**IND3** в [3] представлено доказательство полноты. Ввиду очевидности доказательство непротиворечивости (надежности) опущено. Действительно, при условии отсутствия неопределенных значений доказательство надежности аксиом сводится к сопоставлению связанных кортежей в отношениях. Вопросы возникают при анализе доказательства полноты системы аксиом **IND1**–**IND3**. При доказательстве условия $\Sigma \vDash \sigma \Rightarrow \Sigma \vdash \sigma$: если зависимость σ выполнима (логически следует из Σ), то σ выводима из Σ с использованием аксиом **IND1**–**IND3**, рассматривается правило (*Rule*) формирования представления БД, которое удовлетворяет зависимостям Σ. Затем показано, что если в БД выполнена зависимость σ, то она выводима из Σ. При такой схеме доказательства необходимо показать выводимость σ для любого состояния БД, а не только единственного состояния, построенного по правилу. Несомненно, можно показать выводимость σ для любого состояния БД, но это в работе [3] не сделано. Во всех остальных работах, посвященных зависимостям включения, доказательство полноты системы аксиом сводится к ссылке на работу [3]. Это была одна из причин написания данной статьи.

Разносторонний анализ нетипизированных зависимостей включения показал, что полная аксиоматизация существует по отдельности для зависимостей включения и функциональных зависимостей, тогда как совместно для этих зависимостей

полная аксиоматизация отсутствует [3–5]. В частных случаях, в том числе для одноместных зависимостей включения и произвольных функциональных зависимостей, существует полная аксиоматизация [6, 7].

Продолжением проблемы взаимодействия зависимостей является построение соответствующих нормальных форм (IDNF), основанных на нормальной форме Бойса–Кодда и с ограничением в виде зависимостей включения для ациклических схем БД [7, 8]. Впервые условия ацикличности схем БД были исследованы в [9]. Структурная интерпретация зависимостей включения в виде графа представлена в [10]. Хотя исследуются отличные от [8] условия ацикличности, в работе [11] показана связь между ними.

Отсутствие полной аксиоматизации говорит о том, что в общем случае нельзя определить выводимость той или иной зависимости. Как следствие, процедура построения нормальных форм в общем случае не имеет решения. На наш взгляд, причиной тому является использование нетипизированных зависимостей включения. Действительно, что означает ссылочная целостность между двумя (и более) неоднородными атрибутами? Формально это нетривиальная функциональная зависимость, которая могла быть реализована при проектировании в структуре логических записей на схеме БД, а не в виде ссылочной целостности. Необходимость использования нетипизированных зависимостей включения в этом случае отпадает. С другой стороны, типизированным зависимостям включения соответствуют тривиальные функциональные зависимости, которые при формировании структуры логических записей не используются. Следовательно, типизированные зависимости включения с нетривиальными функциональными зависимостями не взаимодействуют и могут рассматриваться отдельно друг от друга.

Значительная часть работ посвящена разработке и исследованию алгоритмов поиска зависимостей включения. В работе [10] представлен полиномиальный по времени алгоритм для поиска избыточных зависимостей на основе графического теоретического подхода. Представленный алгоритм является полным аналогом алгоритма поиска избыточных функциональных зависимостей. В работах [11–14] представлены различные алгоритмы поиска зависимостей включения, основанные на использовании свойств схемы БД, анализе данных и обработке входящих запросов. Все эти алгоритмы не гарантируют корректность и полноту обнаружения зависимостей включения, но позволяют частично автоматизировать этот процесс.

Полная автоматизация построения зависимостей включения в общем виде нереализуема, поскольку они отражают специфику бизнес-правил в конкретной прикладной области. Однако способствовать их построению позволяют алгоритмы, которые определяют полноту условий за счет определения пороговых значений качества [16, 17].

В настоящее время активно развивается направление анализа зависимостей включения. Для улучшения и автоматизации проверки и диагностики соблюдения правил на основе описания семантики данных (бизнес-данных) предлагаются структуры и алгоритмы для обнаружения возможных нарушений ограничений бизнес-данных, в том числе при модификации схемы БД [17–20]. Это подчеркивает актуальность проводимых исследований в настоящее время и в будущем.

В данной работе рассматриваются основы формальной теории для типизированных зависимостей включения. Хотя такие зависимости считаются частным случа-

ем нетипизированных зависимостей включения, система аксиом получилась иной: во второй аксиоме отсутствует необходимость использования перестановок. Сами по себе перестановки в системе **IND1-IND3** являются искусственными. Действительно, в технологиях БД атрибуты идентифицируются по имени, а не по расположению. Тогда как перестановки фиксируют расположение атрибута. Вместо перестановок можно использовать переименование атрибутов и т.п., что лишний раз подтверждает искусственную природу нетипизированных зависимостей включения.

Наличие неопределенных значений в БД является неизбежным. Поэтому формальная теория должна учитывать их при проектировании. В работах [21, 22] рассматривается проблема выводимости совместно для функциональных зависимостей и зависимостей включения, допускающих наличие неопределенных значений. Как и в случае отсутствия неопределенных значений, удается построить полную и надежную аксиоматику только в частных случаях. Основным препятствием для получения более общих результатов является взаимодействие функциональных зависимостей и нетипизированных зависимостей включения. В работе [23] рассматриваются простые и частичные нетипизированные зависимости включения с неопределенными значениями, имеющие место в стандарте языка SQL. Представлены две системы аксиом, содержащие аксиомы с перестановками атрибутов. Для обеих систем аксиом утверждается наличие полноты со ссылкой на работу [21], однако в работе [21] при утверждении полноты системы аксиом ссылка идет уже на упомянутую работу [3]. В большинстве рассмотренных работ предполагается, что зависимости включения являются нециклическими.

2. Основы формальной теории

В определении 1 представлена формулировка для типизированных зависимостей включения без учета неопределенных значений. Рассмотрим расширение этого понятия. Предварительно определим соответствующие друг другу кортежи при наличии неопределенных значений.

Определение 2. *Кортеж* $t_i[X]$ ***соответствует*** *кортежу* $t_j[X]$ *по атрибутам* X ($t_j[X] \preccurlyeq t_i[X]$), *если* $t_i[A_l] \neq Null$, *тогда* $t_j[A_l] = t_i[A_l]$ *или* $t_j[A_l] = Null$; *если* $t_i[A_l] = Null$, *тогда* $t_j[A_l] = Null$ *для любого атрибута* $A_l \in X$.

Очевидно, что заданное в определении 2 отношение $t_j[X] \preccurlyeq t_i[X]$ является транзитивным. То есть справедливо утверждение: если $t_j[X] \preccurlyeq t_i[X]$ и $t_i[X] \preccurlyeq t_m[X]$, тогда $t_j[X] \preccurlyeq t_m[X]$.

Определение 3. *Зависимость включения* $\sigma = R_j[X] \subseteq R_i[X]$ *от главной таблицы* $R_i[X]$ *к подчиненной таблице* $R_j[X]$ *по атрибутам* X ***существует***, *если для любого кортежа* $t_j[X] \in R_j[X]$ *имеется соответствующий кортеж* $t_i[X]$ *в отношении* $R_i[X]$. *Такую зависимость будем называть типизированной с допущением неопределенных значений.*

Замечание. Кортеж $t_j[X] \in R_j[X]$ может иметь множество соответствующих кортежей в отношении $R_i[X]$. Для замены неопределенных значений в кортеже $t_j[X]$ могут быть выбраны только значения одноименных атрибутов одного из соответствующих кортежей отношения $R_i[X]$, как это было показано в примере 2.

Обозначим множество зависимостей включения, определенных на схеме БД через Σ, а σ пусть будет произвольная зависимость, возможно, σ является элементом множества Σ.

Определение 4. *Зависимость σ является логическим следствием множества зависимостей Σ ($\Sigma \models \sigma$), если данные в БД удовлетворяют всем зависимостям в Σ, тогда данные удовлетворяют зависимости σ. В этом случае зависимость σ будем называть* ***выполнимой****.*

Заметим, что все зависимости Σ по определению 4 являются выполнимыми.

Представим систему аксиом, для зависимостей включения с возможными неопределенными значениями:

- **INN1**) (рефлексивность): если $X \subseteq [R_i]$, тогда $R_i[X] \subsetneq R_i[X]$;
- **INN2**) (проекция): если $R_j[Y] \subsetneq R_i[Y]$ и $X \subseteq Y$, тогда $R_j[X] \subsetneq R_i[X]$;
- **INN3**) (транзитивность): если $R_j[X] \subsetneq R_i[X]$ и $R_i[X] \subsetneq R_l[X]$, тогда выполнено $R_j[X] \subseteq R_l[X]$.

Отличие системы аксиом **INN1**–**INN3** от системы **IND1**–**IND3**, кроме допущения неопределенных значений, в отсутствии перестановок в аксиоме **IND2**. В типизированных зависимостях включения могут быть сопоставлены друг другу только одноименные атрибуты, а на каком они находятся месте – не важно. Это соответствует существующим технологиям БД: на логическом уровне атрибуты идентифицируются своим именем, а не своей позицией в наборе значений.

Заметим, что аксиомы **INN1**–**INN3** задают правила вывода.

Определение 5. *Зависимость σ* ***выводима*** *из Σ за счет системы аксиом ($\Sigma \vdash \sigma$), если при применении аксиом к зависимостям Σ за конечное число шагов будет получена зависимость σ.*

Для любой системы аксиом, прежде всего, необходимо показать ее непротиворечивость (надежность). Для этого покажем, что если зависимость σ выводима из множества Σ с использованием системы аксиом, то она является логическим следствием Σ: $\Sigma \vdash \sigma \Rightarrow \Sigma \models \sigma$.

Теорема 1 (Надежность)**.** *Система аксиом* **INN1**–**INN3** *надежна.*

Доказательство. Последовательно докажем надежность каждой из аксиом.

Рефлексия. Рассмотрим произвольный кортеж $t \in R_i$. В соответствии с определениями 2 и 3 кортеж t всегда будет соответствовать сам себе для любого $X \subseteq [R_i]$, что доказывает надежность аксиомы **INN1**.

Проекция. Предположим, что зависимость $R_j[X] \subsetneq R_i[X]$ не выполнена. Тогда допустимы реализации R_j и R_i такие, что существует кортеж $t_j \in R_j$, для которого нет соответствующего кортежа в отношении R_i по атрибутам X. Поскольку выполнена зависимость $R_j[Y] \subsetneq R_i[Y]$, для кортежа t_j существует соответствующий кортеж $t_i \in R_i$ по атрибутам Y, то есть $t_j[Y] \preccurlyeq t_i[Y]$. Поскольку $X \subseteq Y$, то $t_j[X] \preccurlyeq t_i[X]$, что противоречит предположению об отсутствии соответствующего кортежа для t_j по атрибутам X. Это доказывает надежность аксиомы **INN2**.

Транзитивность. Рассмотрим произвольный кортеж $t_j \in R_j$. Поскольку выполнена зависимость $R_j[X] \subsetneq R_i[X]$, то существует кортеж $t_i \in R_i$: $t_j[X] \preccurlyeq t_i[X]$. Так как имеет место зависимость $R_i[X] \subsetneq R_l[X]$, то существует кортеж $t_l \in R_l$ такой, что $t_i[X] \preccurlyeq t_l[X]$. В силу транзитивности операции $\preccurlyeq$ выполнено условие $t_j[X] \preccurlyeq t_l[X]$, что доказывает надежность аксиомы **INN3**. Теорема доказана.

Замечание. Для отношений, удовлетворяющих условию аксиомы транзитивности, выполнены следующие соотношения:

$$R_j[X] \cap R_i[X] \subsetneq R_i[X] \cap R_l[X],$$
$$R_j[X] \cap R_i[X] \subsetneq R_j[X] \cap R_l[X],$$
$$R_j[X] \cap R_l[X] \subsetneq R_i[X] \cap R_l[X],$$

где $\cap$ – реляционный оператор пересечения.

В [1] после доказательства надежности системы аксиом функциональных зависимостей рассматриваются правила (теоремы), которые позволяют сократить вывод других правил и получать полезные свойства схемы БД. Наиболее полезными являются правила декомпозиции и объединения функциональных зависимостей. Аналогом правила декомпозиции является аксиома проекции **INN2**, а аналога правила объединения для зависимостей включения не существует: из зависимости $R_j[X \cup Y] \subsetneq R_i[X \cup Y]$ выводимы зависимости $R_j[X] \subsetneq R_i[X]$ и $R_j[Y] \subsetneq R_i[Y]$. Обратное утверждение, к сожалению, не верно. Это является принципиальным отличием функциональных зависимостей от зависимостей включения. С одной стороны, это упрощает доказательство полноты системы аксиом **INN1**–**INN3**, с другой стороны – отсутствует возможность удаления атрибутов в зависимостях при построении их минимального покрытия (рассмотрено далее).

Рассмотрим оригинальное доказательство полноты системы аксиом **INN1**–**INN3**, в котором учтены ранее высказанные замечания.

Теорема 2 (Полнота)**.** *Система аксиом* **INN1**–**INN3** *полна.*

Доказательство. Необходимо показать, что если зависимость $\sigma = R_j[X] \subsetneq R_i[X]$ выполнима: $\Sigma \vDash \sigma$, то она выводима: $\Sigma \vdash \sigma$.

Для того, чтобы зависимость σ была выводима, достаточным условием является существование цепочки выполнимых зависимостей:

$$\begin{cases} R_j[Y_1] \subsetneq R'_1[Y_1] \\ R'_1[Y_2] \subsetneq R'_2[Y_2] \\ \qquad\vdots \\ R'_k[Y_{k+1}] \subsetneq R_i[Y_{k+1}], \end{cases} \tag{1}$$

где $X \subseteq Y_l$, $l = \overline{1, k+1}$. Действительно, по аксиомам проекции (**INN2**) и транзитивности (**INN3**), а в случае $i = j$ и по аксиоме рефлексии (**INN1**), получаем выводимость зависимости σ.

Предположим, что зависимость σ не выводима. Тогда любая последовательность (1) содержит, по крайней мере, одну не выполнимую зависимость, пусть это будет $\sigma' = R'_{m-1}[Y_m] \subsetneq R'_m[Y_m]$, где $Y_m = X$, и/или зависимость σ' выполнима, но $X \not\subseteq Y_m$. В этом случае в R'_{m-1}, а следовательно, в R_j, может существовать кортеж t, которому нет соответствующего кортежа в R'_m, а следовательно, в R_i, по атрибутам X.

Получили, что зависимость σ не выполнима. Допустим, что в Σ есть зависимости, которые препятствуют появлению кортежа t в R'_{m-1} по атрибутам X. Поскольку это должно выполняться для любого состояния БД, удовлетворяющего Σ, то зависимость $R'_{m-1}[X] \subsetneq R_i[X]$ является выполнимой и ее можно использовать вместо невыполнимого участка последовательности (1). Полученные противоречия доказывают теорему.

Заметим, что предложенную схему доказательства достаточно просто распространить на нетипизированные зависимости включения. Однако по выше названным причинам данный вид зависимостей в нашей статье не рассматривается.

Аксиомы **INN1**–**INN3** задают правила вывода для зависимостей включения. Следовательно, они могут быть использованы для поиска выводимых (избыточных) зависимостей в Σ.

3. Минимальное покрытие множества зависимостей

На практике ссылочная целостность, теоретической основой которой являются зависимости включения, реализуется СУБД в виде индексных файлов. Эти файлы надо хранить и модифицировать в процессе работы СУБД, что требует дополнительной памяти и времени. Поэтому целесообразно избавиться от избыточных зависимостей в Σ.

В предыдущем разделе доказано, что выводимая зависимость является выполнимой. Следовательно, ее можно удалить без всяких нежелательных последствий для БД: множество допустимых состояний БД останется без изменений. Поиск выводимых зависимостей напрямую является экспоненциальной задачей, поэтому воспользуемся известным аппаратом построения замыканий [1] для функциональных зависимостей. Аналогичный аппарат без доказательства корректности использован в работе [10]. Адаптируем эти результаты для типизированных зависимостей включения.

Определение 6. *Замыканием отношения R_i на множестве зависимостей Σ относительно атрибутов X будем называть множество отношений $R_i^+[X]$, где $R_j \in R_i^+[X]$, если зависимость $\sigma = R_j[X] \subsetneq R_i[X]$ выводима из Σ за счет аксиом* **INN1**–**INN3**, *то есть $\Sigma \vdash \sigma$.*

Рассмотрим алгоритм построения замыкания. Текущее замыкание обозначим $R_i^*[X]$. Будем считать, что используемые в алгоритме множества имеют глобальные имена и их не надо передавать в процедуру через параметры.

PROCEDURE CLOSURE($R_i^*[X]$);
$R_i^*[X] = \emptyset$;
IF $X - [R_i] \neq \emptyset$ THEN EXIT PROC;
$R_i^*[X] = R_i$;
substitution = TRUE;
WHILE *substitution*;
  substitution = FALSE;
  FOR EACH $R_l[Y] \subsetneq R_m[Y]$ FROM Σ;

IF $R_m \in R_i^*[X]$ AND $R_l \notin R_i^*[X]$ AND $X \subseteq Y$ THEN;
$R_i^*[X] = R_i^*[X] \cup R_l$; *substitution* = TRUE;
END IF;
END FOR;
END WHILE;
END PROC;

Внешний цикл WHILE не имеет явного ограничения. Однако для выполнения следующего цикла необходимо дополнение хотя бы одного отношения к замыканию во внутреннем цикле FOR. Следовательно, максимальное количество итераций в алгоритме равно nk, где n – количество зависимостей в Σ и k – количество отношений в БД.

Теорема 3 (Замыкание)**.** *Алгоритм* CLOSURE *корректно формирует множество* $R_i^+[X]$.

Доказательство. Пусть R_j – произвольное отношение и X произвольное множество атрибутов. Необходимо показать, что $R_j \in R_i^+[X]$ тогда и только тогда, когда $R_j \in R_i^*[X]$, или $R_i^+[X] = R_i^*[X]$.

1. (Необходимость) Пусть $R_j \in R_i^*[X]$. Множество $R_i^*[X]$ формируется в трех операторах:

а) $R_i^*[X] = \emptyset$. Множество $R_i^*[X]$ остается пустым, если X содержит атрибуты, которых нет в отношении R_i. По аксиомам **INN1**–**INN3** в этом случае также ничего не выводимо: $R_i^+[X] = \emptyset$.

б) $R_i^*[X] = R_i$. При выполнении условия $X \subseteq [R_i]$ замыкание $R_i^+[X]$ также будет содержать R_i по аксиоме рефлексии **INN1**.

в) $R_i^*[X] = R_i^*[X] \cup R_l$, если выполнены условия $R_l[Y] \subsetneq R_m[Y] \in \Sigma$, $R_m \in R_i^*[X]$, $R_l \notin R_i^*[X]$ и $X \subseteq Y$. По индукции покажем, что $R_l \in R_i^+[X]$. Базис индукции соответствует варианту (б). Предположим, что все отношения в $R_i^*[X]$ до появления зависимости $R_l[Y] \subsetneq R_m[Y]$ соответствуют выводимым зависимостям, то есть зависимость $R_m[X] \subsetneq R_i[X]$ выводима. Тогда по аксиоме проекции **INN2** имеем $R_l[X] \subsetneq R_m[X]$, так как $X \subseteq Y$, и по аксиоме транзитивности **INN3** имеем $R_l[X] \subsetneq R_i[X]$. Это верно для любого l, где $R_l \in R_i^*[X]$, в том числе для $l = j$. Следовательно, $R_i^*[X] \subseteq R_i^+[X]$.

2. (Достаточность) Пусть $R_j \in R_i^+[X]$. Тогда существует k строк вывода, где последней строкой является зависимость $R_j[X] \subsetneq R_i[X]$. При $k = 1$ имеем $j = i$, либо $R_j[X] \subsetneq R_i[X] \in \Sigma$. В обоих случаях по алгоритму отношение R_j будет присоединено к $R_i^*[X]$. Пусть условие выполнимости имеет место для всех зависимостей, вывод которых содержит не более $k-1$ строк. Кроме вариантов, рассмотренных для случая $k = 1$, зависимость $R_j[X] \subsetneq R_i[X]$ может быть получена за счет аксиомы проекции **INN2** из зависимости $R_j[Y] \subsetneq R_i[Y]$, где $X \subseteq Y$. Поскольку R_i уже содержится в $R_i^*[X]$, и, если R_j еще нет в $R_i^*[X]$, то по алгоритму R_j будет присоединено к $R_i^*[X]$, поскольку выполнены все три условия оператора IF.

Кроме того, зависимость $R_j[X] \subsetneq R_i[X]$ может быть получена за счет аксиомы транзитивности **INN3** и R_j пока нет в $R_i^*[X]$. Тогда должно существовать отношение R_l, что зависимость $R_l[Y] \subsetneq R_i[Y]$, где $X \subseteq Y$, выводима, и зависимость $R_j[Z] \subsetneq R_l[Z]$, где $X \subseteq Z$, принадлежит Σ. По предположению $j \neq l$. Тогда

$R_l \in R_i^*[X]$, так как цепочка вывода зависимости $R_l[Y] \subsetneq R_i[Y]$ короче, чем k. Поскольку вновь выполнены все условия оператора IF, то R_j будет принадлежать $R_i^*[X]$. Следовательно, $R_i^+[X] \subseteq R_i^*[X]$. Теорема доказана.

Имея в распоряжении полиномиальный алгоритм поиска избыточных зависимостей, осталось воспользоваться им для построения не избыточного множества типизированных зависимостей включения. Такое множество в [1] называется минимальным покрытием.

PROCEDURE MIN-COVER(Σ);
FOR EACH $\{R_j[X] \subsetneq R_i[X]\}$ FROM Σ;
  IF $R_j \in$ CLOSURE($R_i^*[X]$) THEN;
    $\Sigma = \Sigma - \{R_j[X] \subsetneq R_i[X]\}$;
  END IF;
END FOR;
END PROC;

С учетом количества итераций в алгоритме CLOSURE, результирующее количество итераций в алгоритме MIN-COVER будет равно n^2k.

После рассмотрения алгоритма MIN-COVER закономерным является вопрос об эквивалентности зависимостей Σ на входе и на выходе алгоритма. Однако уже доказанная связь между замыканием и выводимостью и то, что все выводимые зависимости являются выполнимыми, гарантирует одни те же ограничения на допустимые состояния БД со стороны зависимостей включения до и после работы алгоритма MIN-COVER.

4. Заключение

В работе рассмотрены типизированные зависимости включения, которые, по мнению авторов, наиболее приемлемы при классическом подходе к проектированию БД [1, 2]. Количественное соотнесение значений атрибутов друг с другом, прежде всего, определяется функциональными зависимостями. Тогда как задача зависимостей включения – определять количественное соотнесение объектов друг с другом. Использование нетипизированных зависимостей включения приводит к смешению этих двух базовых видов зависимостей и, как следствие, к проблемам при проектировании схемы БД. Причем эти проблемы сказываются не только в теории, но и на практике, когда объект БД подменяется связью на схеме.

Для подтверждения сказанных слов рассмотрим пример схемы БД на сайте разработчиков MySQL: https://dev.mysql.com/doc/sakila/en/. Очевидно, что при формировании этой схемы использовались в основном эвристики, а зависимости на данных имели второстепенное значение или игнорировались вовсе. Поэтому схема содержит множество неточностей. Рассмотрим одну из них, касающуюся содержания нашей статьи. Отношение (таблица) “film” имеет в своем составе атрибуты: “language_id” и “original_language_id”. Для этих атрибутов установлена ссылочная целостность с отношением “language” по атрибуту “language_id”. В результате имеем

нетипизированную зависимость включения:

$$\text{film[original\_language\_id]} \subsetneq \text{language[language\_id]}.$$

Все фильмы имеют значение атрибута “язык оригинала”, однако только некоторые из них имеют перевод. Могут быть еще атрибуты: “автор перевода”, “дата выпуска перевода” и т.д. Не будем вдаваться в детали корректного проектирования схемы БД на основе зависимостей, это займет много места. Отметим только, что отношение БД было склеено с отношением “film”. В схеме должно существовать отдельное отношение “translate” с типизированными зависимостями включения:

$$\text{translate [language \_ id]} \subsetneq \text{language[language \_ id]}$$

и

$$\text{translate[film\_id]} \subsetneq \text{film[film\_id]},$$

а нетипизированная зависимость должна быть удалена вместе с соответствующим атрибутом.

Отметим, что при рассмотрении формальной теории зависимостей включения в нашей работе не потребовалось ограничиваться ациклическими зависимостями и ограничивать арность зависимостей, что делает предложенный аппарат достаточно универсальным. На практике циклические зависимости могут существенно снизить функциональные возможности БД, например при дополнении новой информацией. Однако это проблема проектировщика БД, который формирует множество Σ, и эта проблема решается на семантическом уровне. В качестве технологической поддержки предложенный аппарат может быть расширен правилами и алгоритмами поиска циклических зависимостей.

Список литературы / References

[1] Ullman J., *Principles of database systems*, Computer Science Press, Stanford University, 1980, 379 pp.

[2] Maier D., *The theory of relational databases*, Computer Science Press, Rockville, 1983, 637 pp.

[3] Casanova M., Fagin R., Papadimitriou C., “Inclusion Dependencies and Their Interaction with Functional Dependencies”, *Journal of Computer and System Sciences*, **28**:1 (1984), 29–59.

[4] Chandra A. K., Vardi M. Y., “The Implication Problem for Functional and Inclusion Dependencies is Undecidable”, *SIAM Journal on Computing*, **14**:3 (1985), 671–677.

[5] Fagin R., Vardi M. Y., “Armstrong databases for functional and inclusion dependencies”, *Information Processing Letters*, **16**:1 (1983), 13–19.

[6] Kanellakis P. C., Cosmadakis S. S., Vardi M. Y., “Unary inclusion dependencies have polynomial time inference problems”, Proceedings of the fifteenth annual ACM symposium on Theory of computing (STOC ’83) (New York, USA, 1983), 1983, 264–277.

[7] Cosmadakis S. S., Kanellakis P. C., Vardi M. Y., “Polynomial-time implication problems for unary inclusion dependencies”, *J. ACM*, **37**:1 (1990), 15–46.

[8] Levene M., Vincent M. W., “Justification for Inclusion Dependency Normal Form”, *IEEE Transactions on Knowledge and Data Engineering*, **12**:2 (2000), 281–291.

[9] Beeri C., Fagin R., Maier D., Yannakakis M., "On the Desirability of Acyclic Database Schemes", *ACM*, **38**:3 (1983), 479–513.

[10] Missaoui R., Godin R., "The Implication Problem for Inclusion Dependencies: A Graph Approach", *SIGMOD Record*, **19**:1 (1990), 36–40.

[11] Зыкин В. С., "Ссылочная целостность данных в корпоративных информационных системах", *Информатика и ее применения*, **9**:3 (2015), 119–127; [Zykin V. S., "Ssylochnaya tselostnost dannykh v korporativnykh informatsionnykh sistemakh", *Informatika i ee primeneniya*, **9**:3 (2015), 119–127, (in Russian).]

[12] Biskup J., Dublish P., "Objects in Relational Database Schemes with Functional, Inclusion and Exclusion Dependencies", *Theoretical Informatics and Applications*, **27** (1993), 183–219.

[13] Johnson D. S., Klug A., "Testing Containment of Conjunctive Queries under Functional and Inclusion Dependencies", *Computer and System Sciences*, **28** (1984), 167–189.

[14] Marchi F. D., Lopes S., Petit J. M., "Efficient Algorithms for Mining Inclusion Dependencies", Advances in Database Technology - EDBT 2002 (Prague, Czech Republic, 2002), 2002, 199–214.

[15] Ma S, Fan W, Bravo L., "Extending inclusion dependencies with conditions", *Theoretical Computer Science*, **515** (2014), 64–95.

[16] Bauckmann J., Abedjan Z., Leser U., Müller H., Naumann F., "Discovering conditional inclusion dependencies", 21st ACM international conference on Information and knowledge management (CIKM '12) (ACM, New York, NY, USA), 2012, 2094–2098.

[17] Gómez-López F. T., Gasca R. M., Pérez-Álvarez J. M., "Compliance validation and diagnosis of business data constraints in business processes", *Information Systems*, **48** (2015), 26–43.

[18] Visser J., "Coupled Transformation of Schemas, Documents, Queries, and Constraints", *Electronic Notes in Theoretical Computer Science*, **200**:3 (2008), 3–23.

[19] Garmany J., Walker J., Clark T., *Logical Database Design Principles*, CRC Press, Auerbach Publications, New York, NY, USA, 2005, 69 pp.

[20] Lopes S., Petit J.M., Toumani F., "Discovering interesting inclusion dependencies: application to logical database tuning", *Information Systems*, **27**:1 (2002), 1–19.

[21] Levene M., Loizou G., "Null Inclusion Dependencies in Relational Databases", *Information and Computation*, **136**:2 (1997), 67–108.

[22] Levene M., Loizou G., "The additivity problem for data dependencies in incomplete relational databases", *Semantics in Databases*, Lecture Notes in Computer Science., **189**, Springer-Verlag Berlin Heidelberg, 1998, 136–169.

[23] Köhler H., Link S., "Inclusion Dependencies Reloaded", The 24th ACM International on Conference on Information and Knowledge Management (CIKM '15) (ACM, New York, NY, USA), 2015, 1361–1370.

Zykin V. S., Zykin S. V., "Analysis of Typed Inclusion Dependencies with Null Values", *Modeling and Analysis of Information Systems*, **24**:2 (2017), 155–167.

DOI: 10.18255/1818-1015-2017-2-155-167

Abstract. Null values have become an urgent problem since the creation of the relational data model. The impact of the uncertainty affects all types of dependencies used in the design and operation of the database. This fully applies to the inclusion dependencies, which are the theoretical basis for referential integrity on the data. Attempts to solve this problem contain inaccuracy in the statement of the problem and its solution. The errors in formulation of the problem can be associated with the use in the definition of untyped inclusion dependencies, which leads to permutations of the attributes, although, the attributes in database technology are identified by name and not by their place. In

addition, linking with the use of the inclusion dependencies of heterogeneous attributes, even of the same type, is a sign of lost functional dependencies and leads to interaction of inclusion dependencies and non-trivial functional dependencies. Inaccuracies in the solution of the problem are contained in the statements of axioms and the proof of their properties, including completeness. In this paper we propose an original solution of this problem only for typed inclusion dependencies in the presence of Null values: a new axiom system is proposed, its completeness and soundness are proved. On the basis of inference rules we developed an algorithm for the construction of a not surplus set of typed inclusion dependencies. The correctness of the algorithm is proved.

Keywords: database, inclusion dependences, axiomatic, NULL values

About the authors:
Vladimir S. Zykin, orcid.org/0000-0002-6492-2464, graduate student,
Omsk State Technical University,
11 Mira av., Omsk 644050, Russia, e-mail: vszykin@mail.ru

Sergey V. Zykin, orcid.org/0000-0002-0576-2149, doctor of sciences in technic, professor,
Sobolev Institute of Mathematics SB RAS,
13 Pevtsova str., Omsk 644043, Russia, e-mail: szykin@mail.ru

Моделирование и анализ информационных систем. Т. 24, №2 (2017), с. 168–185
Modeling and Analysis of Information Systems. Vol. 24, No 2 (2017), pp. 168–185

DOI: 10.18255/1818-1015-2017-2-168-185

УДК 517.9

О бифуркациях при малых возмущениях в логистическом уравнении с запаздыванием

Кащенко С. А.

получена 12 января 2017

Аннотация. В статье рассматриваются бифуркационные задачи для логистического уравнения с запаздыванием при наличии малых возмущений. Наиболее интересны результаты для случая, когда малые возмущения содержат большое запаздывание. В качестве основных результатов получены специальные нелинейные эволюционные нормальной формы уравнения, нелокальная динамика которых определяет поведение решений исходного уравнения в малой окрестности состояния равновесия или цикла. Как оказывается, принципиальное значение имеет порядок величины большого запаздывания. Для наиболее простого случая, когда этот порядок совпадает с величиной, обратной к фигурирующему в уравнении малому параметру, нормальная форма представляет собой комплексное уравнение с запаздыванием. В том случае, когда порядок коэффициента запаздывания еще выше, в качестве нормальной формы выступает многопараметрическое семейство специальных краевых задач вырожденно-параболического типа. Все это позволяет сделать вывод о том, что в рассматриваемых задачах с большим запаздыванием характерно явление мультистабильности.

Ключевые слова: нелинейная динамика, бифуркации, асимптотическое представление

Для цитирования: Кащенко С. А., "О бифуркациях при малых возмущениях в логистическом уравнении с запаздыванием", *Моделирование и анализ информационных систем*, **24**:2 (2017), 168–185.

Об авторах:
Кащенко Сергей Александрович, orcid.org/0000-0002-8777-4302, д-р физ.-мат. наук, профессор,
Ярославский государственный университет им. П.Г. Демидова,
ул. Советская, 14, г. Ярославль, 150003 Россия, e-mail: kasch@uniyar.ac.ru

Введение

Логистическое уравнение с запаздыванием

$$\frac{\partial u}{\partial t} = \lambda[1 - u(t-T)]u \quad (\lambda > 0, \quad T > 0) \tag{1}$$

принадлежит к числу фундаментальных уравнений математической экологии. Исследованию решений этого уравнения посвящена значительная литература [1–8]. Напомним простейшие свойства решений этого уравнения. Через $C_{[-T,0]}$ ниже обозначается пространство непрерывных на отрезке $[-T, 0]$ функций со стандартной

нормой. Это пространство примем в качестве фазового, т. е. пространства начальных условий уравнения (1).

Перечислим ряд утверждений о решениях уравнения (1). Простые доказательства их будем опускать.

1. Для уравнения (1) имеет место теорема существования и единственности решений, т. е. для каждого значения t_0 и каждой начальной функции $\varphi(s) \in C_{[-T,0]}$ при всех $t > t_0$ существует и единственно решение $u(t, \varphi)$ уравнения (1), для которого $u(t_0 + s, \varphi) = \varphi(s)$.

2. При условии $\varphi(s) \geqslant 0$ выполнено неравенство $u(t, \varphi) \geqslant 0$ $(t \geqslant t_0)$, а при условии $\varphi(0) > 0$ имеет место строгое неравенство $u(t, \varphi) > 0$ $(t \geqslant t_0)$. В дальнейшем термин «решение» применяется только к неотрицательным решениям (1) и соответственно начальные функции $\varphi(s)$ предполагаются неотрицательными.

3. Линеаризованное в окрестности состояние равновесия $u \equiv 0$ уравнение (1) имеет вид

$$\dot{u} = \lambda u.$$

Отсюда следует, что при всех $\lambda > 0$ нулевое состояние равновесия неустойчиво.

Уравнение (1) имеет состояние равновесия $u_0 \equiv 1$. Линеаризуем на нем это уравнение. Тогда получим уравнение

$$\dot{v} = -\lambda v(t - T). \tag{2}$$

Его характеристический квазиполином имеет вид

$$\mu = -\lambda \exp(-\mu T). \tag{3}$$

Из отрицательности вещественных частей всех корней этого уравнения следует асимптотическая устойчивость решений (2), а значит, и асимптотическая устойчивость состояния равновесия u_0 уравнения (1). Если же имеется корень (3) с положительной вещественной частью, то решение (2) и решение u_0 в (1) неустойчивы.

4. Для отрицательности вещественных частей всех корней (3) необходимо и достаточно выполнение неравенств

$$0 < \lambda T < \frac{\pi}{2}. \tag{4}$$

Докажем это утверждение. Сначала отметим, что при всех достаточно малых и положительных λ все корни (3) имеют отрицательные вещественные части. Действительно, при $\lambda = 0$ имеется нулевой корень $\mu_0 = 0$ (а вещественные части всех остальных корней «равны» $-\infty$). При малых λ уравнение (3) имеет такой корень $\mu_0(\lambda)$, что $\mu_0(0) = (0)$. Тогда

$$\left.\frac{d\mu_0(\lambda)}{d\lambda}\right|_{\lambda=0} = -1.$$

Отсюда следует, что при всех малых положительных λ для всех корней (3) выполнено неравенство $\operatorname{Re}\mu < 0$.

Заметим, что при $\lambda > 0$ корень уравнения (3) не может обратиться в нуль. Пусть при некотором значении $\lambda = \lambda_0$ уравнение (3) имеет пару чисто мнимых корней $\mu_{1,2}(\lambda_0) = \pm i\sigma$, т. е.

$$i\sigma = -\lambda \exp(-i\sigma T).$$

Отсюда получаем, что

$$\lambda\cos(\sigma T) = 0 \quad \text{и} \quad \sigma = \lambda\sin\sigma T.$$

Из первого уравнения находим, что $\sigma = (\pi n + \pi/2)T^{-1}$ $(n = 0, 1, \ldots)$, а из второго тогда приходим к выводу, что n — четное и $\lambda = \lambda_n$ где $\lambda_n T = \pi n + 1/2$. Таким образом, наименьшее из значений λ_n является $\lambda_0 = \pi(2T)^{-1}$. Отсюда следует, что при всех $\lambda \in (0, \lambda_0)$ все корни (3) имеют отрицательные вещественные части.

Последнее, что осталось заметить для завершения обоснования сформулированного выше утверждения, это тот факт, что при увеличении λ корни (3) могут пересекать мнимую ось, двигаясь только из левой комплексной полуплоскости в правую. Действительно, пусть для некоторого корня $\mu(\lambda)$ выполнено условие $\mu(\lambda^0) = i\omega$, а значит, $i\omega = -\lambda^0\exp(-i\omega T)$. Тогда

$$\operatorname{Re}\frac{d\mu(\lambda)}{d\lambda}\bigg|_{\lambda=\lambda^0} = \omega h[\lambda^0(1+\omega^2T^2)]^{-1} > 0.$$

Утверждение доказано.

5. Отметим еще, что в случае, когда функция $v(t) = u(t) - 1$ является, начиная с некоторого момента t_0, знакопостоянной, то из (1) следует её монотонное (при $t \geqslant t_0 + T$) стремление к нулю при $t \to \infty$. Обратим внимание, что для монотонного стремления к нулю некоторого решения $v(t)$ уравнения (2) необходимо и достаточно, чтобы уравнение (3) имело вещественный отрицательный корень.

Условие существования отрицательного корня в (3) состоит в выполнении неравенств

$$0 < \lambda T \leqslant e^{-1}.$$

6. Уравнение (1) является диссипативным: при достаточно больших t каждое решение $u(t)$ этого уравнения удовлетворяет неравенству

$$u(t) \leqslant \exp(\lambda T).$$

Действительно, если функция $u(t) - 1$, начиная с некоторого момента времени, знакопостоянна, то $u(t)$ стремится к 1 при $t \to \infty$. Таким образом, необходимо рассмотреть лишь те решения, у которых бесконечно много (при $t \to \infty$) корней уравнения $u(t) = 1$. Локальные максимумы решений u_m реализуются через отрезок времени T после обращения $u(t)$ в 1. Поэтому

$$u_m = \exp[\lambda(1 - \int\limits_{t_m-T}^{t_m} u(s)ds)] \leqslant \exp(\lambda T).$$

В [2, 3, 7] исследовался вопрос об отыскании всех тех значений параметров λ и T, при которых состояние равновесия u глобально устойчиво, т. е. все решения (1) стремятся к 1 при $t \to \infty$. В [2] было показано, что область глобальной устойчивости выделяется неравенствами

$$0 < \lambda T \leqslant \frac{37}{24}. \tag{5}$$

В [3, 7] проведен алгоритм, который допускает улучшение этой оценки. Сразу отметим, что с его помощью определяются оценки сверху всех решений (1) даже при отсутствии глобальной устойчивости состояния равновесия.

В [4] показано, что при всех $\lambda T > \pi/2$ уравнение (1) имеет непостоянное периодическое решение. В том случае, когда λT мало отличается от $\pi/2$, применимы стандартные методы теории бифуркаций. В разделе 1 для уравнения (1) рассмотрена классическая задача о бифуркации Андронова — Хопфа. В разделах 2–4 исследуются бифуркационные задачи, возникающие при малых возмущениях уравнения (1), причем основное внимание уделено ситуации, когда возмущающая нелинейная добавка содержит функцию u с большим запаздыванием. При этом в разделах 2 и 3 речь пойдет о локальном анализе в окрестности состояния равновесия, а в разделе 4 – в окрестности цикла. В разделе 5 в качестве примера приведены результаты для комплексного логистического уравнения с запаздыванием.

1. Бифуркация Андронова — Хопфа

При $\lambda = \lambda_0$ и $T = T_0$, где $\lambda_0 T_0 = \pi/2$, характеристическое уравнение (3) имеет пару чисто мнимых корней $\pm i\sigma : \sigma = \pi(2T_0)^{-1}$, а все остальные корни (3) имеют отрицательные вещественные части. Отсюда заключаем, что уравнение (2) имеет периодические решения,

$$V_0(t) = \xi \exp(i\sigma t) + \bar{\xi} \exp(-i\sigma t),$$

где ξ — произвольная комплексная постоянная.

В задаче об устойчивости состояния равновесия $u_0 = 1$ уравнения (1) при этом возникает критический случай пары чисто мнимых корней, т. е. реализуются условия так называемой бифуркации Андронова – Хопфа. Для изучения решений (1) при λ и T, близких соответственно к λ_0 и T_0, применим стандартные методы теории бифуркаций.

Положим в (1)

$$\lambda = \lambda_0 + \varepsilon\lambda_1, \quad T = T_0 + \varepsilon T_1,$$

где λ_1 и T_1 как-то фиксированы, а параметр ε является положительным и достаточно малым:

$$0 < \varepsilon \ll 1.$$

Тогда при всех достаточно малых ε в достаточно малой и не зависящей от ε окрестности состояния равновесия $u_0 = 1$ существует двумерное устойчивое локальное интегральное инвариантное многообразие (см., например, [9, 10]). На этом многообразии уравнение (1) можно записать в виде скалярного комплексного обыкновенного дифференциального уравнения

$$\frac{dg}{dt} = (\varepsilon\alpha_1 + O(\varepsilon^2))g + (d + O(\varepsilon))g|g|^2 + O(|g|^5). \tag{6}$$

Если $\operatorname{Re}\alpha_1 \neq 0$ и $\operatorname{Re} d \neq 0$, то в (6)удобно произвести нормировочные замены $g(t) \to \sqrt{\varepsilon}g$, $\tau \to \varepsilon\tau$. Тогда уравнение (6) с точностью до слагаемых порядка $O(\varepsilon)$ принимает вид

$$\frac{d\xi}{d\tau} = \alpha_1\xi + d\xi|\xi|^2. \tag{7}$$

Решения уравнения (7) и уравнения (1) связаны асимптотической формулой

$$u = 1 + \sqrt{\varepsilon}[\xi(\tau)\exp(i\sigma t) + \bar{\xi}(\tau)\exp(-i\sigma t)] + \\ + \varepsilon u_2(t,\tau) + \varepsilon^{3/2}u_3(t,\tau) + \dots, \tag{8}$$

где функции $u_j(t,\tau)$ являются периодическими с периодом $2\pi/\sigma$ по первому аргументу.

Для того, чтобы найти коэффициенты α_1 и d, а значит, ответить на вопрос о поведении всех решений (1) в окрестности u_0 при достаточно малых ε, подставим формальный ряд (8) в (1) и будем последовательно приравнивать коэффициенты при одинаковых степенях ε в левой и правой частях получившегося формального тождества.

На первом шаге, приравнивая коэффициенты при $\varepsilon^{1/2}$, получаем верное равенство, поскольку функция $V_0(t)$ является решением уравнения (2).

На втором шаге соберем коэффициенты при ε^1. В результате получим уравнение для нахождения $u_2(t,\tau)$:

$$\frac{du_2}{dt} = -\lambda_0 u_2(t-T,\tau) - \lambda_0[\xi^2\exp(2i\sigma - i\sigma T) + \\ + \bar{\xi}^2\exp(-2i\sigma + i\sigma T)].$$

Отсюда получаем, что

$$u_2(t,\tau) = A\xi^2\exp(2i\sigma t) + \bar{A}\bar{\xi}^2\exp(-2i\sigma t)$$

и

$$A = \frac{2-i}{5}.$$

На третьем шаге учитываем коэффициенты при $\varepsilon^{3/2}$. В итоге получаем уравнение для u_3:

$$\dot{u}_3 = -\lambda_0 u_3(t-T,\tau) + A_1\exp(i\sigma t) + A_3\exp(3i\sigma t) + \\ + \bar{A}_1\exp(-i\sigma t) + \bar{A}_3\exp(-3i\sigma t). \tag{9}$$

Значение коэффициента A_3 несущественно, а для A_1 имеет место формула

$$A_1 = \frac{d\xi}{d\tau} - \alpha_1\xi - d|\xi|^2\xi, \tag{10}$$

где

$$\alpha_1 = (1+\frac{\pi^2}{4})^{-1}[(\frac{\pi}{2}+i)\lambda_1 + \lambda_0^2 T_1(1-i\frac{\pi}{2})], \tag{11}$$

$$d = -\lambda_0[3\pi - 2 + i(\pi+6)](10(1+\frac{4}{\pi^2}))^{-1}. \tag{12}$$

Условие разрешимости уравнения (9) в классе $2\pi/\sigma$ -периодических функций состоит в выполнении равенства $A_1 = 0$. Отсюда, с учетом равенств (10)—(12), приходим к итоговому уравнению (7) с найденными коэффициентами α_1 и d. Уравнение (7) интегрируется в явном виде. Важно отметить, что

$$\operatorname{Re} d < 0. \tag{13}$$

Поэтому при условии $\operatorname{Re}\alpha_1 \leqslant 0$ все решения (7) стремятся к нулю при $\tau \to \infty$. Если же $\operatorname{Re}\alpha_1 > 0$, то уравнение (7) имеет устойчивый цикл

$$\xi_0(\tau) = \xi_0 \exp(i\varphi_0\tau),$$

в котором

$$\xi_0 = [10(\frac{\pi}{2}\lambda_1 + \lambda_0^2 T_1)(3\pi - 2)^{-1}]^{1/2},$$
$$\varphi_0 = \operatorname{Im}\alpha_1 + \xi_0^2 \operatorname{Im} d.$$

Сформулируем итоговый результат.

Теорема 1. *Пусть* $\operatorname{Re}\alpha_1 > 0$. *Тогда при всех достаточно малых* ε *уравнение* (6) *имеет устойчивый цикл* $u_0(t,\varepsilon)$, *для которого имеет место асимптотическое представление*

$$u_0(t,\varepsilon) = 1 + \sqrt{\varepsilon}(\xi_0(\tau)\exp(i\sigma t) + \bar{\xi}_0(\tau)\exp(-i\sigma t)) + \varepsilon u_2(t,\varepsilon) + O(\varepsilon^{3/2}). \tag{14}$$

2. Бифуркация в окрестности состояния равновесия в случае, когда малые возмущения содержат большое запаздывание

Рассматриваются логистические уравнения с запаздыванием и с малым возмущением

$$\dot{u} = \lambda[1 - u(t-T)]u + \varepsilon F(u, u(t-h)), \tag{15}$$

где некоторая нелинейная функция $F(*,*)$ является достаточно гладкой. Для параметра ε, как и выше, выполнено условие $0 < \varepsilon \ll 1$, поэтому последнее слагаемое в (15) означает малое возмущение уравнения (1).

Сначала остановимся на простейшем случае, когда параметр запаздывания h как-то фиксирован, а для коэффициентов λ и T снова выполнены условия

$$\lambda = \lambda_0 + \varepsilon\lambda_1, \quad T = T_0 + \varepsilon T_1 \quad \text{и} \quad \lambda_0 T_0 = \frac{\pi}{2}. \tag{16}$$

Выделим главные слагаемые функции $F(u, u(t-h))$ в окрестности значений $u \equiv 1$:

$$F(1+V, 1+V(t-h)) = F(1,1) + aV + bV(t-h) + f(V, V(t-h)), \tag{17}$$

где $f(V, V(t-h)$ имеет в нуле порядок малости не ниже второго.

Уравнение (15) имеет положительное состояние равновесия

$$u_0(\varepsilon) = 1 + \varepsilon\lambda_0^{-1}F(1,1) + O(\varepsilon^2). \tag{18}$$

Линеаризуем (15) на $u_0(\varepsilon)$. Характеристический квазиполином получающегося уравнения имеет вид

$$\mu = -(\lambda_0 + \varepsilon\lambda_1)\exp(-\mu(T_0 + \varepsilon T_1)) + \varepsilon a + \varepsilon b\exp(-\mu h). \tag{19}$$

Этот квазиполином имеет два корня $\mu_1(\varepsilon)$ и $\mu_2(\varepsilon)$, близкие при $\varepsilon \to 0$ к мнимой оси:

$$\mu_{1,2} = \pm i\sigma + O(\varepsilon), \tag{20}$$

а все остальные его корни имеют отрицательные вещественные части, которые отделены от нуля при $\varepsilon \to 0$. Тем самым применимы все результаты предыдущего раздела. Используя в (15) формальное разложение (8), приходим к уравнению на двумерном локальном инвариантном интегральном многообразии

$$\frac{d\xi}{d\tau} = (\alpha_1 + (a+b)(1 - i\frac{\pi}{2})(1 + \frac{\pi^2}{4})^{-1})\xi + d\xi|\xi|^2, \tag{21}$$

где коэффициенты α_1 и d те же, что и в (7). По решениям уравнения (21) с помощью формулы (8) восстанавливаются решения уравнения (15) на рассматриваемом двумерном многообразии и формулируются стандартые (см. Теорему 1) выводы о существовании и устойчивости периодического решения (15).

Более интересна ситуация, когда параметр h в (15) является достаточно большим. Пусть для некоторого фиксированного значения h_1 имеем

$$h = \frac{h_1}{\varepsilon}. \tag{22}$$

Для состояния равновесия $u_0(\varepsilon)$ верна формула (18), а структура корней характеристического квазиполинома (19) меняется существенным образом. Дело в том, что уже бесконечно много корней в (19) стремятся к мнимой оси при $\varepsilon \to 0$. Покажем это.

Положим в (19) $\mu = i\sigma + \varepsilon\mu_1$. Тогда для $\mu_1 = \mu_0 + O(\varepsilon)$ приходим к уравнению

$$\begin{aligned}\mu_1 = [1 - \frac{\pi}{2}\exp(-i\sigma T)]^{-1}[a - (\lambda_1 - i\sigma\lambda_0 T_1)\exp(-i\sigma T_0) + \\ + b\exp(i\varphi(\varepsilon))\cdot\exp(-\mu_1 h_1)].\end{aligned} \tag{23}$$

Здесь $\varphi(\varepsilon) = (-\sigma h_1)\varepsilon^{-1}|_{mod 2\pi}$, $\varphi(\varepsilon) \in [0, 2\pi)$. При $\varepsilon \to 0$ выражение $\varphi(\varepsilon)$ бесконечно много раз изменяется от 0 до 2π.

Уравнение (23), очевидно, имеет счетное множество корней. Отсюда уже просто следует вывод о том, что уравнение (19) имеет бесконечно много корней, которые стремятся к мнимой оси при $\varepsilon \to 0$. Это означает, что в задаче об устойчивости состояния равновесия $u_0(\varepsilon)$ уравнения (15) при условии (22) реализуется критический случай бесконечной размерности. Такого рода критические случаи изучались в работах автора [11, 12].

Применим результаты из [11] для уравнения (15). Снова рассмотрим формальный ряд (3). Подставим его в (15) и будем собирать коэффициенты при одинаковых степенях ε, считая, что величина $\varphi = \varphi(\varepsilon)$ фиксирована. Повторяя предыдущие построения, для неизвестной амплитуды $\xi(\tau)$ приходим к уравнению с фиксированным запаздыванием

$$\begin{aligned}\frac{d\xi}{d\tau} = (\alpha_1 + a(1 - i\frac{\pi}{2})(1 + \frac{\pi^2}{4})^{-1})\xi + \\ + b(1 - i\frac{\pi}{2})(1 + \frac{\pi^2}{4})^{-1}\exp(-\frac{i\sigma h_1}{\varepsilon})\xi(\tau - h_1) + d|\xi|^2\xi,\end{aligned} \tag{24}$$

где $\tau = \varepsilon t$. Таким образом аналогом (21) здесь является уравнение (24) в бесконечномерном фазовом пространстве.

Сформулируем основные утверждения, обоснования которых вытекают из приведенных выше построений. Фиксируем произвольно $\varphi_0 \in [0, 2\pi)$ и рассмотрим уравнение

$$\frac{d\xi}{d\tau} = (\alpha_1 + a)\xi + b\exp(i\varphi_0)\xi(\tau - h_1) + d|\xi|^2\xi, \tag{25}$$

где $a = (1 - i\pi/2)(1 + \pi^2/4)^{-1}$, $b = b(1 - i\pi/2)(1 + \pi^2/4)^{-1}$.

Теорема 2. *Пусть при некотором φ_0 уравнение (25) имеет ограниченное при $\tau \to \infty$ решение $\xi(\tau)$. Тогда существует последовательность $\varepsilon_n \to 0$, определяемая равенством $\varphi(\varepsilon) = \varphi_0$, что при $\varepsilon = \varepsilon_n$ уравнение (15) имеет асимптотическое по невязке с точностью до $O(\varepsilon^{3/2})$ решение*

$$\begin{aligned} u = 1 + \sqrt{\varepsilon}_n(\xi(\tau)\exp(i\sigma t) + \bar{\xi}(\tau)\exp(-i\sigma t)) + \\ + \varepsilon_n[\lambda_0^{-1}F(1,1) + \frac{2-i}{5}\xi^2(\tau)\exp(2i\sigma t)] + \\ + \frac{2+i}{5}\xi^{-2}(\tau)\exp(-2i\sigma t)], \quad \tau = \varepsilon_n t. \end{aligned}$$

Для простейших решений уравнения (25) вида

$$\xi_0(\tau) = \rho_0\exp(i\varphi_0\tau) \quad (\rho_0 > 0) \tag{26}$$

можно получить более точные утверждения.

Итак, пусть для некоторого φ_0 уравнение (25) имеет, при выполнении некоторых условий типа общности положения, устойчивое (неустойчивое) решение (26). Тогда при достаточно малых ε_n уравнение (15) имеет устойчивое (неустойчивое) периодическое решение

$$u_0(t, \varepsilon) = 1 + \sqrt{\varepsilon}_n 2\rho_0\cos((\sigma + \varepsilon_n\varphi_0 + O(\varepsilon_n^2))t) + O(\varepsilon_n).$$

Отметим, что количество решений вида (26) уравнения (25) тем больше, чем больше значение параметров b и h_1. Таким образом, условие (22) больших значений запаздывания может приводить к существенному усложнению динамических свойств в окрестности состояния равновесия (15).

3. О локальной динамике уравнения (15) в случае сверхбольших значений h

Коротко остановимся на ситуации, когда параметр h в (15) является «сверхбольшим», т. е. параметр h_1 в (22) является большим:

$$h_1 = \gamma^{-1} \quad \text{и} \quad 0 < \gamma \ll 1. \tag{27}$$

Повторяя предыдущие построения, получаем уравнение (24). Ниже считаем, что параметр ε в (24) как-то фиксирован.

Ограничимся исследованием решений (24) из малой, но не зависящей от ε и γ окрестности нулевого состояния равновесия. Удобно в (24) произвести замену времени $\tau_1 = \gamma^{-1}\tau$. Тогда получим сингулярно возмущенное уравнение

$$\gamma\frac{d\xi}{d\tau_1} = (\alpha_1 + a)\xi + b\exp(\frac{i\sigma}{\varepsilon\gamma})\xi(\tau_1 - 1) + d|\xi|^2\xi. \tag{28}$$

Характеристический квазиполином для линеаризованного в нуле уравнения (28) имеет вид

$$\gamma\mu = A_1 + A_2\exp(-\frac{i\sigma}{\varepsilon\gamma})\exp(-\mu), \tag{29}$$

где $A_1 = \alpha_1 + (1 - i\pi/2)(1 + \pi^2/4)^{-1}a$, $A_2 = (1 - i\pi/2)(1 + \pi^2/4)^{-1}b$. Положим $A_1 = A_{11} + \varepsilon A_{12}$ и $A_{20} = |A_2| = |b|(1 + \pi^2/4)^{-1/2}$. Тогда величина A_2 представима в виде $A_2 = A_{20}\exp(i\varphi_{20})$.

При условии $A_{11} > 0$ квазиполином (29) имеет корень с положительной и отделенной от мнимой оси при $\gamma \to 0$ вещественной частью. Поэтому задача о динамике (28) в окрестности нуля становится нелокальной. Ниже предполагаем, что

$$A_{11} < 0. \tag{30}$$

Далее, при условии $|A_1|A_{20}^{-1} < 1$ тоже получаем, что у квазиполинома (29) есть корень с положительной вещественной частью, равной $\ln A_{20}|A_1|^{-1}$. Отметим, что при условии $|A_1|A_{20}^{-1} > 1$ есть корень (29) с отрицательной вещественной частью, однако для отрицательности вещественных частей всех корней (29) этого условия недостаточно. Сформулируем итоговое утверждение о корнях (29).

Лемма 1. *Пусть выполнено неравенство* (30) *и пусть*

$$A_{20} < |A_{11}|. \tag{31}$$

Тогда при всех достаточно малых γ все корни (29) *имеют отрицательные и отделенные от нуля при $\gamma \to 0$ вещественные части. Если же*

$$A_{20} > |A_{11}|, \tag{32}$$

то при достаточно малых γ найдется корень (29) *с положительной и отделенной от нуля при $\gamma \to 0$ вещественной частью.*

Простое обоснование этой леммы опустим.

Таким образом, в изучении нуждается только случай, когда значение A_{20} близко к $|A_{11}|$. В связи с этим ниже полагаем, что для некоторого фиксированного A_0 выполнено равенство

$$A_2 = |A_{11}| + \gamma^2 A_0. \tag{33}$$

В этом случае бесконечно много корней (29) стремятся к мнимой оси при $\gamma \to 0$, а все остальные корни имеют отрицательные и отделенные от нуля при $\gamma \to 0$ вещественные части. Таким образом, опять реализуется критический случай (в задаче об устойчивости стационара $u_0(\varepsilon)$) бесконечной размерности. Метод изучения таких критических случаев разработан в [13–16]. Применим здесь соответствующие результаты.

Сначала исследуем поведение корней характеристического уравнения (29) при $\gamma \to 0$. Найдем асимптотику тех корней, которые стремятся к мнимой оси при $\gamma \to 0$. Введем несколько обозначений. Положим в (29) $\mu = i\omega$ и результат запишем в виде

$$P(\omega) = A_{20} \exp(i\varphi_2 - i\omega),$$

где $P(\omega) = i(\gamma\omega - A_{12}) - A_{11}$. Имеем

$$\min_{\omega} |P(\omega)| = |P(\omega_0)| = A_{11} \quad \text{и} \quad \omega_0 = A_{12}\gamma^{-1}.$$

Введем еще одно обозначение. Через $\varkappa = \varkappa(\gamma)$ обозначим такое значение из полуинтервала $[0, 2\pi)$, для которого величина

$$\sigma(\varepsilon\gamma)^{-1} + A_{12}\gamma^{-1} - \varphi_{20} + \varkappa(\gamma)$$

является целой кратной 2π. Сформулируем итоговое утверждение.

Лемма 2. *При условиях* (27), (30), *и* (33) *уравнение* (29) *имеет бесконечно много корней* $\mu_n(\gamma)$ $(n = 0, \pm1, \pm2, \ldots)$, *для которых имеют место асимптотические представления*

$$\mu_n(\gamma) = i[A(\varepsilon\gamma)^{-1} + A_{12}\gamma^{-1} - \varphi_{20} + \varkappa(\gamma) + 2\pi n] + \gamma\mu_{n_1} + \gamma^2\mu_{n_2} + \ldots, \tag{34}$$

где

$$\mu_{n_1} = -i(2\pi n + \varkappa)A_{20}^{-1},$$
$$\mu_{n_2} = -\frac{1}{2}(2\pi n + \varkappa)^2 A_{20}^{-2} + i(2\pi n + \varkappa)A_{20}^{-2} + A_0.$$

Линейное уравнение

$$\gamma\frac{d\xi}{d\tau_1} = A_1\xi + A_2 \exp(-\frac{i\sigma}{\varepsilon\gamma})\xi(\tau_1 - 1) \tag{35}$$

имеет при условиях (30), (33) бесконечно много решений вида

$$\xi_n(\tau_1, \gamma) = \eta_n \exp(i\mu_n(\gamma)\tau_1), \tag{36}$$

где η_n – произвольные комплексные постоянные.

Формулу (36) запишем иначе

$$\xi_n(\tau_1, \gamma) = \exp(iB(\gamma)\tau_1) \cdot \exp(2\pi ni(1 - \gamma A_{20}^{-1})\tau_1)\eta_n(s), \tag{37}$$

в которой

$$B(\gamma) = -\sigma(\varepsilon\gamma)^{-1} + A_{12}\gamma^{-1} + \varkappa(\gamma)(1 - \gamma A_{20}^{-1}),$$
$$\eta_n(s) = \exp[-(\mu_{n_2} + O(\gamma))s], \quad s = \gamma^2\tau_1.$$

Решение нелинейного уравнения (28) будем искать в виде линейной комбинации всех решений вида (37) с неизвестными амплитудами $\eta_n(s)$:

$$\xi(\tau_1, \gamma) = \gamma \exp(iB(\gamma)\tau_1) \cdot \sum_{n=-\infty}^{\infty} \eta_n(s) \exp(2\pi nix) +$$
$$+ \gamma^{3/2} g_1(\tau_1, x, s) + \ldots, \quad x = (1 - \gamma A_{20}^{-1}). \tag{38}$$

Здесь $g_1(\tau_1, x, s), \ldots$ — периодические по первым двум аргументам функции. Подставим (38) в (28). Производя стандартные действия, получим уравнение для $g_1(\tau_1, x, s)$. Из условия его разрешимости приходим к уравнению для функции

$$\eta(s,x) = \sum_{n=-\infty}^{\infty} \eta_n(s) \exp(2\pi n i x) :$$

$$\frac{\partial \eta}{\partial s} = (2A_{20}^2)^{-1} \frac{\partial^2 \eta}{\partial x^2} + A_{20}^{-2}(i\varkappa + 1)\frac{\partial \eta}{\partial x} + \\ + (A_0 + \varkappa^2 A_{20}^{-2})\eta + A_{20}^{-1} d\eta|\eta|^2. \tag{39}$$

Поскольку функция $\eta(s,x)$ 1-периодически зависит от второго аргумента, уравнение (39) должно быть дополнено периодическими краевыми условиями

$$\eta(s, x+1) \equiv \eta(s,x). \tag{40}$$

Из приведенных построений вытекает следующее утверждение.

Теорема 3. *Пусть выполнены условия* (27), (30) *и* (33). *Пусть для некоторого* $\varkappa = \varkappa_0 \in [0, 2\pi)$ *краевая задача* (39), (40) *имеет ограниченное при* $s \to \infty$, $x \in [0,1]$ *решение* $\eta_0(s,x)$. *Тогда найдется такая последовательность* $\gamma_m \to 0$, *определяемая из условия* $\varkappa(\gamma_m) = \varkappa_0$, *что уравнение* (28) *имеет асимптотическое по невязке с точностью до* $O(\gamma^3)$ *решение*

$$\xi(\tau_1, \gamma) = \gamma \eta(s,x), \quad s = \gamma^2 \tau_1, \quad x = (1 - \gamma A_{20}^{-1})\tau_1, \quad \gamma = \gamma_m.$$

Уравнение (39) с краевыми условиями (40) является классическим уравнением Гинзбурга — Ландау. Его динамике посвящены исследования многих авторов (см., например, [17–21]). Известно, в частности, что это уравнение может иметь довольно сложную, в том числе — нерегулярную динамику.

Отметим, что в краевой задаче (39), (40) просто находятся решения — бегущие волны — вида

$$\eta = \rho \exp(i 2\pi m x + i\varphi s).$$

Для них можно получить результаты о существовании точных решений в (28) и ответить на вопрос об их устойчивости.

Таким образом, в этом разделе показано, что увеличение запаздывания h_1 согласно соотношению (27) может приводить к усложнению динамических свойств исходного уравнения.

Особо отметим, что при различных значениях $\varkappa$ динамика (39), (40) может быть различной. Это означает, что при $\gamma \to 0$ может происходить неограниченный процесс прямых и обратных бифуркаций в (39), (40).

Далее встает вопрос о структуре решений (28) при дальнейшем уменьшении γ (увеличении h_1). На первый взгляд принципиальных изменений не происходит, поскольку параметр γ фигурирует только в выражении $\varkappa = \varkappa(\gamma)$ уравнения (39). Однако это не совсем так. Дело в том, что выше исследования приведены для случая, когда «надкритичность» A_0 связана со значением A_2 формулой (33).

Необходимо, конечно, рассмотреть более общую связь, когда

$$A_2 = |A_{11}| + \gamma^{\delta} A_0, \quad 0 < \delta < 2. \tag{41}$$

Соответствующие результаты были получены в [14, 15]. Здесь лишь укажем, что отличие случаев $\delta = 2$ и $0 < \delta < 2$ является существенным. Например, возникают более сложные многопараметрические семейства нелинейных краевых задач, играющих роль краевой задачи (39), (40). Все это говорит о том, что при $\gamma \to 0$ (в условии (41)) может происходить резкое усложнение динамики. Например, спонтанное увеличение количества установившихся режимов в (28).

4. Малые возмущения в окрестности цикла

Здесь предполагаем, что параметры r и T в (15) фиксированы и $rT > \frac{\pi}{2}$, т. е. в (1) имеется экспоненциально орбитально устойчивый цикл $u_0(t)$ периода $\widetilde{T}_0$.

При условии малости параметра γ приходим к стандартной задаче о малом возмущении грубого цикла. Введем несколько обозначений.

Сначала отметим, что функция $\dot{u}_0(t)$ является решением линеаризованного на $u_0(t)$ уравнения (1)

$$\dot{v} = r(1 - u_0(t - T))v - ru_0(t)v(t - T). \tag{42}$$

Формально сопряженным к этому уравнению является уравнение

$$\dot{y} = -r(1 - u_0(t - T))y + ru_0(t + T)y(t + T). \tag{43}$$

Хейловское скалярное произведение определяется формулой

$$< v(s), y(s) >= v(0)y(0) - r \int_{-T}^{0} u_0(s + 1)v(s)y(s + 1)ds$$

($v(s) \in C_{[-T,0]}$, $y(s) \in C_{[0,T]}$). Отметим, что для любых двух решений (42) и (43), определенных при всех $t \in R$, справедливо тождество

$$< v(s + t), y(s + t) > \equiv < v(s), y(s) > .$$

Уравнение (43) имеет единственное (с точностью до множителя) периодическое решение $y_0(t)$ ($\not\equiv 0$). Положим

$$\sigma =< F(u_0(t), u_0(t - h)), y_0(t) > .$$

Следующее простое утверждение является обобщением хорошо известного для обыкновенных дифференциальных уравнений результата о возмущении грубого цикла.

Теорема 4. *При всех достаточно малых ε уравнение (15) имеет орбитально устойчивый цикл $u_0(t, \varepsilon)$, для которого*

$$u_0(t,\varepsilon) = u_0\left((1+\sigma\varepsilon+o(\varepsilon))t\right) + O(\varepsilon).$$

Более интересна ситуация, когда вместе с условием $0 < \varepsilon \ll 1$ выполнено условие

$$h \gg 1.$$

Результаты о бифуркациях в окрестности цикла при малых возмущениях с большим запаздыванием получены в работах [11, 12]. Применим их к уравнению (15).

Пусть $h = h_1\varepsilon^{-1}$. Введем в рассмотрение формальные ряды

$$u(t,\varepsilon) = u_0(\tau) + \varepsilon u_1(\tau, s) + \dots,$$

$$\frac{d\tau}{dt} = 1 + \varepsilon\varphi(s) + \dots,$$

где $V_j(t)$ – $\widetilde{T}_0$-периодичны по τ, $\varphi(s)$ – скалярная почти периодическая функция, s – «медленное» время: $s = \varepsilon t$. Подставим эти ряды в (15). Тогда, собирая коэффициенты при одинаковых степенях ε, приходим к уравнению

$$\frac{du_1}{d\tau} = r(1 - u_0(\tau - T))u_1 - ru_0u_1(\tau - T) + \varepsilon R(\tau, s),$$

где $R(\tau, s) = F(u_0(\tau), u_0(\tau(t - h_1/\varepsilon)))$. Условие разрешимости для этого уравнения в указанном классе функций состоит в выполнении равенства $< R(\tau, s), y_0(\tau) >= 0$.

Обозначим через $g(z)$ функцию

$$g(z) =< R(u_0(\tau), u_0(\tau(t - z))), y_0(\tau) > . \tag{44}$$

Учитывая, что $\tau(t - h) = \tau(t) - h_1\varepsilon^{-1} - \int\limits_{s-h_1}^{s} (\varphi(s_1) + \dots)ds_1$, из (44) приходим к уравнению для определения $\varphi(s)$:

$$\varphi(s) = g(\Theta + \int\limits_{-h_1}^{0} \varphi(s + p)dp), \tag{45}$$

где $\Theta = \Theta(\varepsilon) = \{\varepsilon^{-1}h_1\}|_{\bmod \widetilde{T}_0}$. После того, как решение $\varphi(s)$ этого уравнения найдено, алгоритм последовательного нахождения коэффициентов фигурирующих выше формальных рядов можно неограниченно продолжать.

Рассмотрим вопрос о состояниях равновесия уравнения (45). Для их нахождения получаем уравнение

$$\varphi = g(\Theta + h_0\varphi).$$

Вопрос об устойчивости некоторого состояния равновесия φ_0 при $\Theta = \Theta_0$ этого уравнения решается стандартным образом.

Теорема 5. *Пусть при некотором $\Theta = \Theta_0$ уравнение (45) имеет состояние равновесия φ_0 и пусть выполнено неравенство*

$$h_1g'(\Theta_0 + h_1\varphi_0) \neq 1.$$

Тогда существует такая последовательность $\varepsilon_n \to 0$, *определяемая из условия* $\Theta(\varepsilon) = \Theta_0$, *что при* $\varepsilon = \varepsilon_n$ *и при достаточно больших* n *уравнение* (15) *имеет периодическое решение* $u_0(t, \varepsilon)$ *вида*

$$u_0(t, \varepsilon) = u_0(\tau) + \varepsilon u_1(\tau, s) + o(\varepsilon),$$

где $\tau = (1 + \varepsilon\varphi_0 + O(\varepsilon^2))$. *Это решение устойчиво (неустойчиво) при*

$$h_0 g'(\Theta_0 + h_0\varphi_0) < 1 \ (> 1).$$

Таким образом, уравнение (45) может иметь любое число (в зависимости от h_1) устойчивых состояний равновесия, а уравнение (15) соответственно такое же число устойчивых периодических решений. Кроме этого, для (45) и (15) характерен неограниченный процесс прямых и обратных бифуркаций при $\varepsilon \to 0$, поскольку $\Theta = \Theta(\varepsilon)$ бесконечно много раз меняется от 0 до $\widetilde{T}_0$ при $\varepsilon \to 0$.

5. Пример. Комплексное логистическое уравнение с запаздыванием

Рассматривается скалярное комплексное уравнение

$$\dot{u} = r[1 - d|u(t - T)|^2]u, \quad d = 1 + i\Delta. \tag{46}$$

Это уравнение имеет периодическое решение

$$u_0(t) = \exp(i\Delta_0 t), \quad \Delta_0 = -r\Delta.$$

Условие устойчивости этого решения состоит в выполнении неравенства

$$rT < \frac{\pi}{4}. \tag{47}$$

Отметим, что после замены $v = |u|^2$ уравнение (46) принимает вид

$$\dot{u} = 2r[1 - v(t - T)]v. \tag{48}$$

Объектами исследования здесь является уравнение (46) с малыми возмущениями с большим запаздыванием

$$\dot{u} = r[1 - d|u(t - T)|^2]u + \varepsilon\alpha u(t - \frac{h_1}{\varepsilon}), \quad \alpha = \alpha_0 \exp(i\psi_0), \tag{49}$$

где $h_1 > 0$, а ε — малый положительный параметр:

$$0 < \varepsilon \ll 1.$$

Ниже в разделе 5.1. исследуется вопрос о локальной (в окрестности $u_0(t)$) динамике уравнения (49).

5.1. О решениях уравнения (49) близких к циклу $u_0(t)$

Уравнение вида (49) изучено в работе [11]. Применим здесь алгоритм исследования из [11]. Для этого введем в рассмотрение формальный асимптотический ряд

$$u = \exp(i\Delta_0\tau)[1 + \varepsilon u_1(\tau) + \ldots], \tag{50}$$

где $\Delta_0 = -r\Delta$, $\tau = \tau(t)$

$$\frac{d\tau}{dt} = 1 + \varepsilon\varphi_0(s) + \ldots, \quad s = \varepsilon t. \tag{51}$$

Отсюда получаем, что

$$\tau(t - \frac{h_1}{\varepsilon}) = \tau(t) - \frac{h_1}{\varepsilon} - \int\limits_{s-h_1}^{s} (\varphi_0(s_1) + \ldots)ds_1. \tag{52}$$

Подставим (50) в (49) и будем собирать коэффициенты при одинаковых степенях ε. Для $u_1(\tau)$ тогда получаем уравнение

$$\begin{aligned} i\Delta_0\varphi_0 + \dot{u}_1 = &-(1 + i\Delta)(u_1(t - T) + \bar{u}_1(t - T)) \\ &+ \exp[-i(\Theta + \Delta_0 \int\limits_{-h_1}^{0} \varphi_0(s + s_1)ds_1)]. \end{aligned} \tag{53}$$

В (53) положено

$$\Theta = \Theta(\varepsilon) = \Delta_0 h_1 \varepsilon^{-1}|_{mod2\pi}.$$

Из условия разрешимости уравнения (53) в классе периодических функций для определения неизвестной функции $\varphi_0(s)$ приходим к уравнению

$$\Delta_0\varphi_0(s) = \sqrt{1 + \Delta_0^2}\sin(\Theta + \Delta_0 \int\limits_{-h_1}^{0} \varphi_0(s + s_1)ds_1 + \varkappa), \quad \varkappa = \operatorname{arctg}(-\Delta_0). \tag{54}$$

Рассмотрим вопрос о состояниях равновесия уравнения (54). Для их нахождения получаем уравнение

$$\varphi = g(\Theta + h_1\varphi), \tag{55}$$

$\varphi = \Delta_0\varphi_0$, $g(x) = \sqrt{1 + \Delta_0^2}\sin(x + \varkappa)$.

Пусть при некотором $\Theta = \Theta_0$ уравнение (55) имеет решение φ^0, т. е. при $\Theta = \Theta_0$ уравнение (54) имеет состояние равновесия $\varphi(s) = \Delta_0\varphi^0$. Исследуем на устойчивость это состояние равновесия. Для этого рассмотрим линеаризованное на φ^0 уравнение (55). В результате приходим к уравнению

$$\psi(s) = \sqrt{1 + \Delta_0^2}\cos(\Theta_0 + h_1\varphi^0) \int\limits_{-h_1}^{0} \psi(s + s_1)ds_1. \tag{56}$$

Имеет место следующее простое утверждение.

Лемма 3. *При условии*

$$h_1\sqrt{1+\Delta_0^2}\cos(\Theta_0+h_1\varphi^0)<1, \tag{57}$$

все корни характеристического квазиполинома уравнения (56) *имеют отрицательные вещественные части, а при*

$$h_1\sqrt{1+\Delta_0^2}\cos(\Theta_0+h_1\varphi^0)>1 \tag{58}$$

имеется положительный корень этого квазиполинома.

Основной результат.

Теорема 6. *Пусть при некотором* $\Theta=\Theta_0$ *уравнение* (55) *имеет состояние равновесия* φ^0 *и пусть выполнено неравенство*

$$h_1\sqrt{1+\Delta_0^2}\cos(\Theta_0+h_1\varphi^0)\neq 1.$$

Тогда существует такая последовательность $\varepsilon_n\to 0$, *определяемая из условия* $\Theta(\varepsilon)=\Theta_0$, *что при* $\varepsilon=\varepsilon_n$ *и при достаточно больших* n *уравнение* (49) *имеет периодическое решение* $u_0(t,\varepsilon)$ *вида*

$$u_0(t,\varepsilon)=\exp(i\Delta_0\tau)[1+\varepsilon u_1(\tau)+O(\varepsilon)],$$

где $\tau=(1+\varepsilon\varphi^0+o(\varepsilon))t$. *Это решение устойчиво при условии* (57) *и неустойчиво при условии* (58).

Из этой теоремы, в частности, следует, что количество устойчивых периодических решений (49) может быть сколь угодно большим в зависимости от величины параметра h_1. Кроме этого, для (49) характерен неограниченный процесс прямых и обратных бифуркаций от цикла $u_0(t)$ при $\varepsilon\to 0$. Это связано с тем, что количество и устойчивость состояний равновесия в (56), а значит, и периодических решений (49), зависит от величины $\Theta=\Theta(\varepsilon)$, которая бесконечно много раз меняется от 0 до 2π при $\varepsilon\to 0$.

Список литературы / References

[1] Wright E. M., “A non-linear difference-differential equation”, *Journal für die reine und angewandte Mathematik*, **194** (1955), 66–87.

[2] Kakutani S., Markus L., “On the non-linear difference-differential equation $y'(t)=(a-by(t-\tau))y(t)$”, *Contributions to the Theory of Nonlinear Oscillations*, **4**, ed. S. Lefschetz, Princeton University Press, Princeton, 1958, 1–18, Annals of Mathematical Studies (AM-41).

[3] К вопросу об оценке в пространстве параметров области глобальной устойчивости уравнения Хатчинсона, *Нелинейные колебания в задачах экологии*, ЯрГУ, Ярославль, 1985, 55–62; [Kashchenko S. A., “K voprosu ob otsenke v prostranstve parametrov oblasti globalnoy ustoychivosti uravneniya Khatchinsona”, *Nelineynye kolebaniya v zadachakh ekologii*, YarGU, Yaroslavl, 1985, 55–62, (in Russian).]

[4] Jones G. S., "The existence of periodic solutions of $f'(x) = -\alpha f(x-1)[1+f(x)]$", *Journal of Contemporary Mathematical Analysis*, **5** (1962), 435–450.

[5] Кащенко С.А., "Сложные стационарные режимы одного дифференциально-разностного уравнения, обобщающего уравнение Хатчинсона", *Исследования по устойчивости и теории колебаний*, ЯрГУ, Ярославль, 1983, 8; [Kashchenko S. A., "Slozhnye statsionarnye rezhimy odnogo differentsialno-raznostnogo uravneniya, obobshchayushchego uravnenie Khatchinsona", *Issledovaniya po ustoychivosti i teorii kolebaniy*, YarGU, Yaroslavl, 1983, 8, (in Russian).]

[6] Кащенко С. А., "О периодических решениях уравнения $x'(t) = -lx(t-1)[1+x(t)]$", *Исследования по устойчивости и теории колебаний*, ЯрГУ, Ярославль, 1978, 110–117; [Kashchenko S. A., "O periodicheskikh resheniyakh uravneniya $x'(t) = -lx(t-1)[1+x(t)]$", *Issledovaniya po ustoychivosti i teorii kolebaniy*, YarGU, Yaroslavl, 1978, 110–117, (in Russian).]

[7] Кащенко С. А., "Асимптотика периодического решения обобщённого уравнения Хатчинсона", *Исследования по устойчивости и теории колебаний*, ЯрГУ, Ярославль, 1981; [Kashchenko S. A., "Asimptotika periodicheskogo resheniya obobshchennogo uravneniya Khatchinsona", *Issledovaniya po ustoychivosti i teorii kolebaniy*, YarGU, Yaroslavl, 1981, (in Russian).]

[8] Kashchenko S., "Asymptotics of the Solutions of the Generalized Hutchinson Equation", *Automatic Control and Computer Sciences*, **47**:7 (2013), 470–494.

[9] Hale J. K., *Theory of functional differential equations*, Springer Verlag, New York, 1977, 626 pp.

[10] Hartman P., *Ordinary Differential Equations*, Wiley, New York, 1965, 626 pp.

[11] Кащенко С. А., "Бифуркации в окрестности цикла при малых возмущениях с большим запаздыванием", *Журнал вычислительной математики и математической физики*, **40**:5 (2000), 693–702; English transl.: Kashchenko S. A., "Bifurcations in the neighborhood of a cycle under small perturbations with a large delay", *Comput. Math. Math. Phys.*, **40**:5 (2000), 659–668.

[12] Kashchenko S. A., "Bifurcational Features in Systems of Nonlinear Parabolic Equations with Weak Diffusion", *International Journal of Bifurcation and Chaos*, **15**:11 (2005), 3595–3606.

[13] Кащенко С. А., "Применение метода нормализации к изучению динамики дифференциально-разностных уравнений с малым множителем при производной", *Дифференциальные уравнения*, **25**:8 (1989), 1448–1451; English transl.: Kashchenko S. A., "Application of the normalization method to the study of the dynamics of a differential-difference equation with a small factor multiplying the derivative", *Differ. Uravn.*, **25**:8 (1989), 1448–1451.

[14] Кащенко И. С., "Асимптотический анализ поведения решений уравнения с большим запаздыванием", *Доклады РАН*, **421**:5 (2008), 586–589; [Kashchenko I. S., "Asymptotic analysis of the behavior of solutions to equations with large delay", *Doklady Mathematics*, **78**:1 (2008), 570–573, (in Russian).]

[15] Кащенко И.С., "Локальная динамика уравнений с большим запаздыванием", *Журнал вычислительной математики и математической физики*, **48**:12 (2008), 2141–2150; English transl.: Kashchenko I. S., "Local dynamics of equations with large delay", *Comput. Math. Math. Phys.*, **48**:12 (2008), 2172–2181.

[16] Кащенко С. А., "Уравнение Гинзбурга — Ландау — нормальная форма для дифференциально-разностного уравнения второго порядка с большим запаздыванием", *Журнал вычислительной математики и математической физики*, **38**:3 (1998), 457–465; English transl.: Kashchenko S. A., "The Ginzburg–Landau equation as a normal form for a second-order difference-differential equation with a large delay", *Comput. Math. Math. Phys.*, **38**:3 (1998), 443–451.

[17] Ахромеева Т. С., Курдюмов С. П., Малинецкий Г. Г., *Нестационарные структуры и диффузионный хаос*, Наука, М., 1992, 544 с.; [Akhromeeva T. S., Kurdyumov S. P., Malinetskiy G. G., *Nestatsionarnye struktury i diffuzionnyy khaos*, Nauka, M., 1992, 544 pp., (in Russian).]

[18] Aranson I. S., Kramer L., "The world of the complex Ginzburg–Landau equation", *Reviews of Modern Physics*, **74**:1 (2002), 99–143.

[19] Кудряшов Н. А., *Методы нелинейной математической физики*, МИФИ, М., 2008, 352 с.; [Kudryashov N. A., *Metody nelineynoy matematicheskoy fiziki*, MIFI, M., 2008, 352 pp., (in Russian).]

[20] Кащенко А. А., "Устойчивость бегущих волн в уравнении Гинзбурга — Ландау с малой диффузией", *Моделирование и анализ информационных систем*, **18**:3 (2011), 58–62; [Kashchenko A. A., "Analysis of running waves stability in the Ginzburg–Landau equation with small diffusion", *Model. Anal. Inform. Sist.*, **18**:3 (2011), 58–62, (in Russian).]

[21] Kashchenko A. A., "Analysis of Running Waves Stability in the Ginzburg–Landau Equation with Small Diffusion", *Automatic Control and Computer Sciences*, **49**:7 (2015), 514–517.

Kashchenko S.A., "About Bifurcations at Small Perturbations in a Logistic Equation with Delay", *Modeling and Analysis of Information Systems*, **24**:2 (2017), 168–185.

DOI: 10.18255/1818-1015-2017-2-168-185

Abstract. The article considers bifurcation problems for a logistic equation with delay at small perturbations. The most interesting results are for the case when small perturbations contain a large delay. The main results are special nonlinear equations of evolution in the normal form. Their nonlocal dynamics defines the behaviour of the solutions of the original equation in a small neigbourhood of the balance state or the cycle. It turns out that the order of large delay magnitude is principal. For the simplest case, when this order is congruent with the magnitude inverse to the small parameter appearing in the equation, the normal form is a complex equation with delay. In the case when the order of the delay coefficient is even higher, the normal form is presented by a multiparameter family of special boundary-value problems of degenerate-parabolic type. All these things allow to make a conclusion about the fact that in the considered problems with large delay the multistability is typical.

Keywords: nonlinear dynamics, bifurcation, asymptotic presentation

About the authors:
Sergey A. Kashchenko, orcid.org/0000-0002-8777-4302, professor,
P.G. Demidov Yaroslavl State University,
14 Sovetskaya str., Yaroslavl 150003, Russia, e-mail: kasch@uniyar.ac.ru

Моделирование и анализ информационных систем. Т. 24, №2 (2017), с. 186–204
Modeling and Analysis of Information Systems. Vol. 24, No 2 (2017), pp. 186–204

DOI: 10.18255/1818-1015-2017-2-186-204

УДК 517.9

Релаксационные циклы в модели синаптически взаимодействующих осцилляторов

Преображенская М. М.

получена 16 января 2017

Аннотация. В настоящей работе рассматривается математическая модель кольцевой нейронной сети с синаптическим взаимодействием элементов. Модель представляет собой систему скалярных нелинейных дифференциально-разностных уравнений, правые части которых зависят от большого параметра. Неизвестные функции, входящие в систему, характеризуют мембранные потенциалы нейронов. Представляет интерес поиск в рамках данной системы уравнений релаксационных циклов, а именно периодических решений с асимптотически большим всплеском на периоде. С этой целью ставится задача отыскания решений в виде дискретных бегущих волн, что позволяет перейти от исследования системы к изучению одного скалярного нелинейного дифференциально-разностного уравнения с двумя запаздываниями. Далее, при стремлении большого параметра к бесконечности определяется предельный объект, представляющий собой релейное уравнение с двумя запаздываниями. Конструктивно, с использованием метода шагов, доказывается, что можно выделить шесть случаев ограничений на параметры, в каждом из которых решение релейного уравнения с начальной функцией из подходящего класса совпадает с одной и той же периодической функцией с требуемыми свойствами. Затем определяется оператор последований Пуанкаре и с использованием принципа Шаудера доказывается существование релаксационного периодического решения сингулярно возмущенного уравнения с двумя запаздываниями. Для этого строится асимптотика этого решения, а затем доказывается его близость к решению релейного уравнения. Из экспоненциальной оценки производной Фреше оператора Пуанкаре следует единственность в построенном классе функций решения дифференциально-разностного уравнения с двумя запаздываниями, а также обосновывается его экспоненциальная орбитальная устойчивость.

Ключевые слова: релаксационные колебания, запаздывание, большой параметр, синаптическая связь

Для цитирования: Преображенская М. М., "Релаксационные циклы в модели синаптически взаимодействующих осцилляторов", *Моделирование и анализ информационных систем*, **24**:2 (2017), 186–204.

Об авторах:
Преображенская Маргарита Михайловна, orcid.org/0000-0002-7032-1155, ассистент,
Ярославский государственный университет им. П.Г. Демидова, ул. Советская, 14, г. Ярославль, 150003 Россия,
младший научный сотрудник, НЦЧ РАН, ул. Лесная, д. 9, г. Черноголовка, Московская область, 142432 Россия,
e-mail: rita.preo@gmail.com

Благодарности:
Работа выполнена при поддержке гранта Российского научного фонда (проект №14-21-00158).

1. Постановка задачи

В настоящей работе используется подход к моделированию химических синапсов, предложенный в статье [1], в основе которого лежит реализация идеи быстрой поро-

говой модуляции. Быстрая пороговая модуляция (fast threshold modulation) — это специальный способ связи динамических систем, для которого характерно скачкообразное изменение правых частей дифференциальных уравнений при переходе некоторых управляющих переменных через свои критические значения (см. [2–7]).

Несколько иная математическая модель цепочки нейронов с синаптической связью была предложена в статье [1] и имеет вид

$$\dot{u_j} = \Big(\lambda f\big(u_j(t-1)\big) + bg\big(u_{j-1}\big)\ln(u_*/u_j)\Big)u_j,\ j = 1,\dots,m,\ u_0 = u_m. \tag{1}$$

Здесь $u_j(t) > 0$ — нормированные мембранные потенциалы нейронов, связанных в кольцо, $\lambda \gg 1$ — большой параметр, характеризующий скорость протекания электрических процессов, $b = \mathrm{const} > 0$, $u_* = \exp(c\lambda)$ — пороговое значение, $c = \mathrm{const} \in \mathbb{R}$, слагаемые $bg\big(u_{j-1}\big)\ln(u_*/u_j)u_j$ моделируют синаптическое взаимодействие. Относительно функций $f(u)$, $g(u)$ предполагаем, что они из класса $C^2(\mathbb{R}_+)$, где $\mathbb{R}_+ = \{u \in \mathbb{R} : u \geqslant 0\}$, и удовлетворяют условиям:

$$\begin{gathered} f(0) = 1;\ f(u) + a,\ uf'(u),\ u^2f''(u) = O\big(u^{-1}\big) \text{ при } u \to +\infty,\ a = \mathrm{const} > 0; \\ g(u) > 0\ \forall u > 0,\ g(0) = 0;\ g(u) - 1,\ ug'(u),\ u^2g''(u) = O\big(u^{-1}\big) \text{ при } u \to +\infty. \end{gathered} \tag{2}$$

Будем искать периодическое решение системы (1) такое, что функции u_j будут иметь один всплеск на периоде с разностью фаз, равной $\Delta = \mathrm{const} > 0$.

Причины, описанные в статье [1], по которым для исследования выбрана система (1), состоят в следующем. Во-первых, связующие слагаемые $bg(u_{j-1})u_j\ln(u_*/u_j)$ меняют знак с «+» на «−» при увеличении потенциалов u_j и при прохождении их через критическое значение u_*. Во-вторых, для системы (1) удается корректно определить предельный объект, которым оказывается некоторая релейная система с запаздыванием, что будет продемонстрировано во втором разделе настоящей статьи.

Анализ сингулярно возмущенной системы (1) основан на следующих двух математических идеях. Первая из них описывается в [1,8–14] и связана с переходом в (1) к логарифмической шкале, то есть с заменой $x_j := (1/\lambda)\ln u_j$, кроме того, вместо большого параметра λ вводится в рассмотрение малый параметр $\varepsilon := 1/\lambda \ll 1$. Эта замена позволяет перейти к близкой к релейной системе

$$\dot{x_j} = F\big(x_j(t-1),\varepsilon\big) + b(c - x_j)G\big(x_{j-1},\varepsilon\big),\ j = 1,\dots,m,\ x_0 = x_m, \tag{3}$$

где $F(x,\varepsilon) := f\big(\exp(x/\varepsilon)\big)$, $G(x,\varepsilon) := g\big(\exp(x/\varepsilon)\big)$.

Вторая идея состоит в поиске периодического решения системы (3) в виде дискретной бегущей волны. Этот способ представления решения сформулирован, например, в статьях [1, 12, 15, 16]. Основная идея состоит в следующей замене переменных:

$$x_j = x\big(t + (j-1)\Delta,\varepsilon\big),\ j = 1,\dots,m, \tag{4}$$

которая приводит к задаче о поиске периодического решения следующего уравнения с двумя запаздываниями

$$\dot{x} = F\big(x(t-1),\varepsilon\big) + b(c - x)G\big(x(t-\Delta),\varepsilon\big). \tag{5}$$

Период решения уравнения (5) должен быть равен $T = m\Delta/k,\ k \in \mathbb{N}$, что следует из условия $x_0 = x_m$.

Для полученного уравнения (5) задача состоит в следующем. Необходимо подобрать параметры a, b, c, Δ, такие, что при всех достаточно малых ε уравнение (5) будет иметь экспоненциально орбитально устойчивый цикл $x = x_*(t,\varepsilon)$ периода $T_*(\varepsilon)$, где

$$\lim_{\varepsilon\to 0} T_*(\varepsilon) = T_* > 0.$$

При этом требуем, чтобы функция $x_*(t,\varepsilon)$ на отрезке времени длины периода имела один промежуток положительности и один промежуток отрицательности. Это с учетом сделанной экспоненциальной замены и будет означать, что функции u_j обладают одним всплеском на периоде с разностью фаз Δ.

2. Анализ вспомогательного уравнения

Из свойств (2) функций f и g следует, что

$$\lim_{\varepsilon\to 0} F(x,\varepsilon) = R(x),\quad \lim_{\varepsilon\to 0} G(x,\varepsilon) = H(x),$$
$$R(x) := \begin{cases} 1, & \text{при } x < 0,\\ -a, & \text{при } x > 0,\end{cases}\quad H(x) := \begin{cases} 0, & \text{при } x < 0,\\ 1, & \text{при } x > 0.\end{cases} \tag{6}$$

Далее, исследуем предельное релейное уравнение

$$\dot{x} = R\big(x(t-1)\big) + b(c-x)H\big(x(t-\Delta)\big), \tag{7}$$

для чего определим класс начальных функций. Так же, как в работах [1, 13, 14], фиксируем постоянные $\sigma_0 > 0$, $q_1 > \sigma_0$, $q_2 \in (0,\sigma_0)$, оценки на которые будут уточнены позднее, и обозначим через $S(\sigma_0,q_1,q_2)$ замкнутое, ограниченное и выпуклое множество функций $\varphi(t)$ (см. рис. 1), определенное следующим образом:

$$S(\sigma_0,q_1,q_2) = \{\varphi \in C[-1-\sigma_0,-\sigma_0] :\ -q_1 \leqslant \varphi(t) \leqslant -q_2\ \forall t \in [-1-\sigma_0,-\sigma_0],\ \varphi(-\sigma_0) = -\sigma_0\}. \tag{8}$$

Через $x_\varphi(t),\ t \geqslant -1-\sigma_0$, обозначим решение уравнения (7) с произвольной начальной функцией $\varphi(t)$, удовлетворяющей (8).

Будем интересоваться периодическим решением $x_\varphi(t)$. Обозначим период через T_φ и дополнительно предположим, что на интервале $(0,T_\varphi)$ функция $x_\varphi(t)$ имеет ровно один ноль t_φ.

Построим решение методом шагов. Отметим, что в зависимости от знака $x(t-1)$ и $x(t-\Delta)$ уравнение (7) принимает одну из четырех форм:

$$\dot{x} = 1 \text{ при } x(t-1) < 0,\ x(t-\Delta) < 0; \tag{A}$$

$$\dot{x} = -a \text{ при } x(t-1) > 0,\ x(t-\Delta) < 0; \tag{B}$$

$$\dot{x} = 1 + b(c-x) \text{ при } x(t-1) < 0,\ x(t-\Delta) > 0; \tag{C}$$

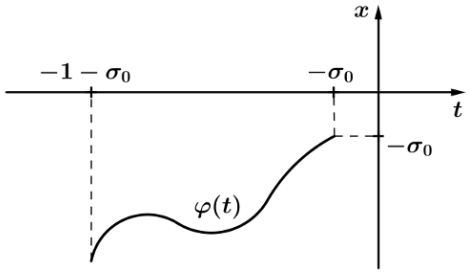


Рис. 1. Вид начальной функции $\varphi(t)$, удовлетворяющей (8)
Fig. 1. The form of the initial function $\varphi(t)$ satisfying (8)

$$\dot{x} = -a + b(c - x) \text{ при } x(t-1) > 0,\ x(t-\Delta) > 0. \tag{D}$$

Обозначим через $x_A(\widetilde{t}, \widetilde{x}; t)$ решение задачи Коши для уравнения из случая (A) с начальным условием $x\big|_{t=\widetilde{t}} = \widetilde{x}$, где $\widetilde{t},\ \widetilde{x}$ — некоторые известные константы. Таким образом, получаем в каждом случае решение соответствующей задачи Коши:

(A): $x_A(\widetilde{t}, \widetilde{x}; t) \equiv t - \widetilde{t} + \widetilde{x}$;

(B): $x_B(\widetilde{t}, \widetilde{x}; t) \equiv -at + a\widetilde{t} + \widetilde{x}$;

(C): $x_C(\widetilde{t}, \widetilde{x}; t) \equiv (\widetilde{x} - 1/b - c)\exp\big(-b(t - \widetilde{t})\big) + 1/b + c$;

(D): $x_D(\widetilde{t}, \widetilde{x}; t) \equiv (\widetilde{x} + a/b - c)\exp\big(-b(t - \widetilde{t})\big) - a/b + c$.

В начале рассмотрим случай, когда

$$\text{I.}\quad 0 < \Delta < 1. \tag{9}$$

На первом этапе рассмотрим отрезок $[-\sigma_0, \Delta]$. На этом промежутке аргументы $x(t-1)$, $x(t-\Delta)$ функций R и H совпадают с функциями $\varphi(t-1)$, $\varphi(t-\Delta)$, которые принимают отрицательные значения, следовательно, здесь имеем дело с задачей Коши (A) при $\widetilde{t} = \widetilde{x} = -\sigma_0$. Таким образом,

$$x_\varphi(t) = t \text{ при } t \in [-\sigma_0, \Delta]. \tag{10}$$

Отметим, что по построению $x_\varphi(0) = 0$, а значит, в силу периодичности выполняется $x_\varphi(T_\varphi) = 0$. Таким образом, с учетом (10) и предположения о том, что t_φ — единственный корень уравнения $x_\varphi(t)$ на интервале $(0, T_\varphi)$, функция $x_\varphi(t)$ устроена так, как показано на рисунке 2, то есть положительна на интервале $(0, t_\varphi)$, а на (t_φ, T_φ) — отрицательна. Кроме того, поскольку речь идет о периодическом решении $x_\varphi(t)$, то дополнительно требуем, во-первых, чтобы выполнялось условие

$$T_\varphi - t_\varphi > 1, \tag{11}$$

а во-вторых, точка T_φ, отвечающая длине периода, должна попадать на промежуток, где решение описывается формулой $x_A(\widetilde{t}, \widetilde{x}; t)$.

Дальнейшее построение решения $x_\varphi(t)$ зависит от взаимного расположения на оси Ot точек Δ, 1, $t_\varphi + \Delta$, $t_\varphi + 1$ переключения релейных функций $R\big(x(t-1)\big)$, $H\big(x(t-\Delta)\big)$ и первых двух нулей t_φ, T_φ решения $x_\varphi(t)$. Что касается значений Δ, 1, $t_\varphi + \Delta$, $t_\varphi + 1$, то, принимая во внимание предположение (9), для них возможны две ситуации:

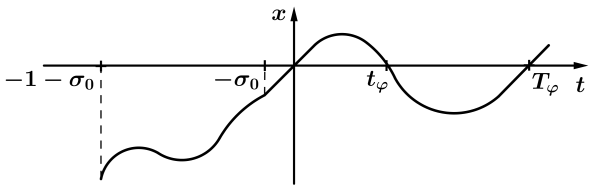


Рис. 2. Предполагаемый вид решения $x_\varphi(t)$ уравнения (7) с начальной функцией, удовлетворяющей (8)
Fig. 2. An assumed form of the solution $x_\varphi(t)$ of the equation (7) with the initial function satisfying (8)

I.1. когда интервалы $(\Delta, 1)$, $(t_\varphi + \Delta, t_\varphi + 1)$ не пересекаются (рис. 3(a)),

I.2. и когда пересекаются (рис. 3(b)).

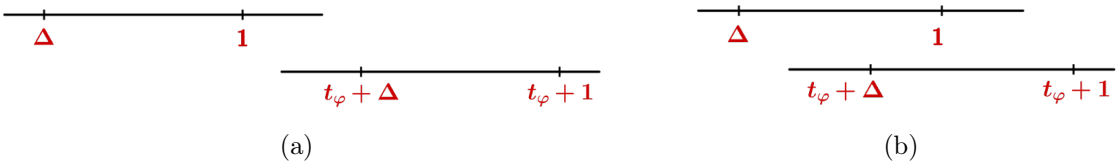


Рис. 3. Случаи взаимного расположения точек переключения релейных функций $R\big(x(t-1)\big)$ и $H\big(x(t-\Delta)\big)$
Fig. 3. The cases of the relative location of the switching points of the relay functions $R\big(x(t-1)\big)$ and $H\big(x(t-\Delta)\big)$

В первой ситуации t_φ может попасть на один из двух участков: $(\Delta, 1)$, $(1, t_\varphi + \Delta)$. Тогда с учетом возможного расположения значения T_φ получаем пять вариантов расположения соответствующих точек:

I.1.1.1 $t_\varphi \in (\Delta, 1)$, $T_\varphi \in (1, t_\varphi + \Delta)$ (рис. 4(a));

I.1.1.2 $t_\varphi \in (\Delta, 1)$, $T_\varphi \in (t_\varphi + \Delta, t_\varphi + 1)$ (рис. 4(c));

I.1.1.3 $t_\varphi \in (\Delta, 1)$, $T_\varphi > t_\varphi + 1$ (рис. 4(e));

I.1.2.1 $t_\varphi \in (1, t_\varphi + \Delta)$, $T_\varphi \in (t_\varphi + \Delta, t_\varphi + 1)$ (рис. 4(g));

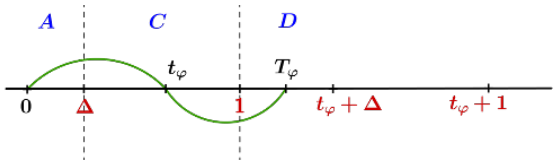
I.1.2.2 $t_\varphi \in (1, t_\varphi + \Delta)$, $T_\varphi > t_\varphi + 1$ (рис. 4(i)).

Во второй ситуации t_φ может принадлежать только интервалу $(\Delta, t_\varphi + \Delta)$, а T_φ — одному из трех промежутков. Таким образом, добавляются еще три случая:

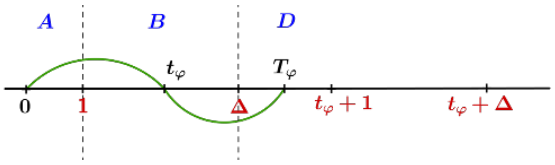
I.2.1 $t_\varphi \in (\Delta, t_\varphi + \Delta)$, $T_\varphi \in (t_\varphi + \Delta, 1)$ (рис. 4(k));

I.2.2 $t_\varphi \in (\Delta, t_\varphi + \Delta)$, $T_\varphi \in (1, t_\varphi + 1)$ (рис. 4(m));

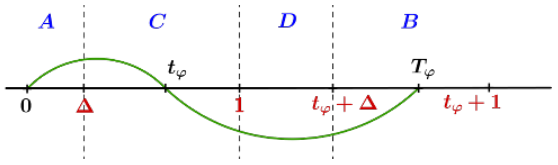
I.2.3 $t_\varphi \in (\Delta, t_\varphi + \Delta)$, $T_\varphi > t_\varphi + 1$ (рис. 4(o)).



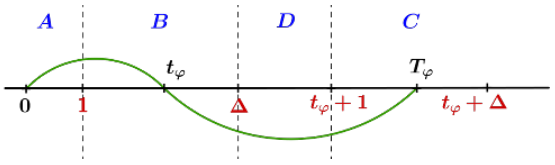
(a) I.1.1.1



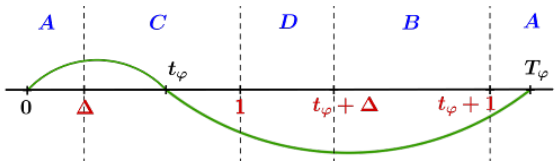
(b) II.1.1.1



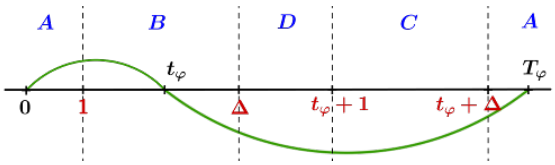
(c) I.1.1.2



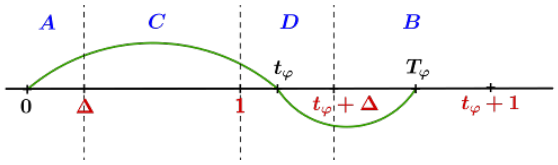
(d) II.1.1.2



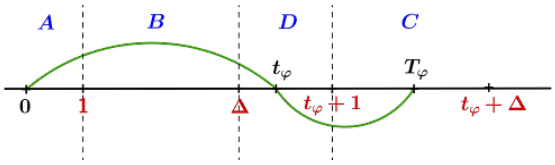
(e) I.1.1.3



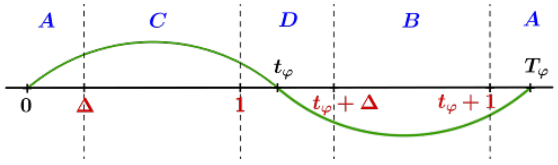
(f) II.1.1.3



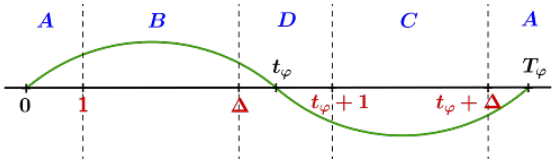
(g) I.1.2.1



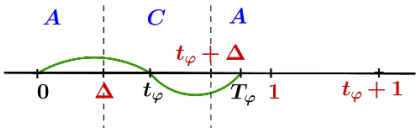
(h) II.1.2.1



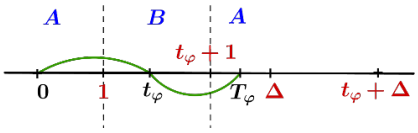
(i) I.1.2.2



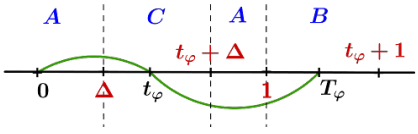
(j) II.1.2.2



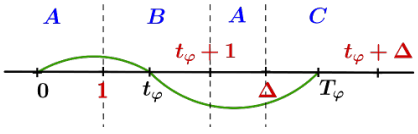
(k) I.2.1



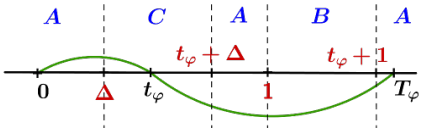
(l) II.2.1



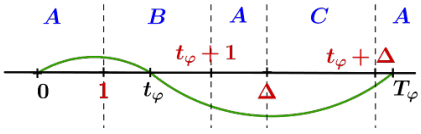
(m) I.2.2



(n) II.2.2



(o) I.2.3



(p) II.2.3

Рис. 4. Случаи взаимного расположения значений 1, Δ, $t_\varphi+1$, $t_\varphi+\Delta$, t_φ, T_φ
Fig. 4. The cases of the relative location of the values 1, Δ, $t_\varphi+1$, $t_\varphi+\Delta$, t_φ, T_φ

Если предположение (9) заменить обратным

$$\text{II.}\quad 1 < \Delta, \tag{12}$$

то получим еще восемь возможных случаев расположения точек переключения релейных функций и нулей решения, аналогичных приведенным (рис. 4(b), 4(d), 4(f), 4(h), 4(j), 4(l), 4(n), 4(p)). Отметим, что в этой ситуации в качестве множества $S(\sigma_0, q_1, q_2)$ начальных функций следует рассматривать класс

$$S(\sigma_0, q_1, q_2) = \{\varphi \in C[-\Delta - \sigma_0, -\sigma_0] :\ -q_1 \leqslant \varphi(t) \leqslant -q_2\ \forall t \in [-\Delta - \sigma_0, -\sigma_0], \\ \varphi(-\sigma_0) = -\sigma_0\}, \tag{13}$$

аналогичный (8), но определенный на отрезке $[-\Delta - \sigma_0, -\sigma_0]$. Связано это с тем, что длина отрезка, на котором заданы начальные функции, совпадает с наибольшим запаздыванием рассматриваемого уравнения. При этом предположение (11) заменяется на

$$T_\varphi - t_\varphi > \Delta. \tag{14}$$

Далее, остается выбрать из полученных шестнадцати случаев те, в которых функция $x_\varphi(t)$ является T_φ-периодической. Это означает, что выполняется условие (11) (в случае I) или (14) (в случае II) и T_φ находится как решение уравнения $x_A(\widetilde{t}, \widetilde{x}; t) = 0$, при некоторых начальных условиях $\widetilde{t} > 0, \widetilde{x} < 0$. Оба эти условия соблюдаются в случаях I.1.1.3, I.1.2.2, I.2.3, II.1.1.3, II.1.2.2, II.2.3. В каждом случае наложим на параметры a, b, c, Δ ограничения, обеспечивающие требуемый порядок точек переключения и нулей решения $x_\varphi(t)$, необходимый знак функции $x_\varphi(t)$ в точках переключения, подходящую монотонность участков решения, заданных экспонентами. Ниже перечислены ограничения на параметры для каждого случая.

I.1.1.3

$$0 < \Delta < 1, \quad \frac{1}{b} + c < 0,$$

$$1 - \Delta < \frac{1}{b} \ln\left(\frac{bc - b\Delta + 1}{b\ c + 1}\right) + \Delta < 1, \quad \exp(b\Delta - b)\ \Big(-\frac{1}{b} - c + \Delta\Big) + \frac{1}{b} + c < 0,$$

$$\frac{(bc+1)\big((a+1)\exp b - (bc - b\Delta + 1)\exp(b\Delta)\big)\exp(-2b\Delta)}{b(bc - b\Delta + 1)} + c < \frac{a}{b};$$

I.1.2.2

$$0 < \Delta < 1, \quad c < \frac{a}{b},$$

$$b\exp(b\Delta)\big(\exp(b\Delta) - 1\big)(bc - a) < 0, \quad \exp\big(b(\Delta - 1)\big)\Big(-\frac{1}{b} - c + \Delta\Big) + \frac{1}{b} + c > 0;$$

I.2.3

$$0 < \Delta < 1, \quad \frac{1}{b} + c < 0$$

$$\Delta < \frac{1}{b} \ln\left(\frac{bc - b\Delta + 1}{b\ c + 1}\right) + \Delta < 1 - \Delta, \quad b(bc+1)\exp(b\Delta)\ \big(\exp(b\Delta) - 1\big) < 0,$$

$$b\big(c\exp(-b\Delta) - c + 2\ \Delta - 1\big) + \ln\left(\frac{bc - b\Delta + 1}{bc + 1}\right) + \exp(-b\Delta) - 1 > 0;$$

II.1.1.3
$$\frac{1}{a}+1<\Delta<\frac{1}{a}+2,$$
$$\exp\left(b\left(-\frac{1}{a}+\Delta-2\right)\right)\left(a\Big(\frac{1}{b}-\Delta+1\Big)-c+1\right)+c<\frac{a}{b},$$
$$-(1+a)\ \exp\left(\Big(2+\frac{1}{a}\Big)b\right)+(1+bc)\exp\left(b\left(1+\frac{1}{a}+\Delta\right)\right)+$$
$$+\exp(b\Delta)\big(b-bc+a(1+b-b\Delta)\big)<0;$$

II.1.2.2
$$1<\Delta<\frac{1}{a}+1,\quad c<\frac{a}{b},$$
$$(\exp b-1)\,(bc-a)<0,\quad bc\big(\exp(b\Delta)-1\big)+\exp(b\Delta)-\exp b(a+1)+a<0;$$

II.2.3
$$\frac{1}{a}+2<\Delta<a+\frac{1}{a}+2,$$
$$\frac{1}{b}+c<\exp\left(-\frac{(a+1)b}{a}\right)\Big(a+\frac{1}{a}+2+\frac{1}{b}+c-\ \Delta\Big).$$

В качестве доказательства совместности условий в каждом случае приведем пример набора констант, удовлетворяющих данным условиям:

I.1.1.3 $\Delta=0.7,\ a=4,\ b=30,\ c=-0.3;$

I.1.2.2 $\Delta=0.7,\ a=9,\ b=6,\ c=-0.1;$

I.2.3 $\Delta=0.35,\ a=4,\ b=20,\ c=-1;$

II.1.1.3 $\Delta=2,\ a=3/2,\ b=8,\ c=-2;$

II.1.2.2 $\Delta=3,\ a=1/4,\ b=5,\ c=-2;$

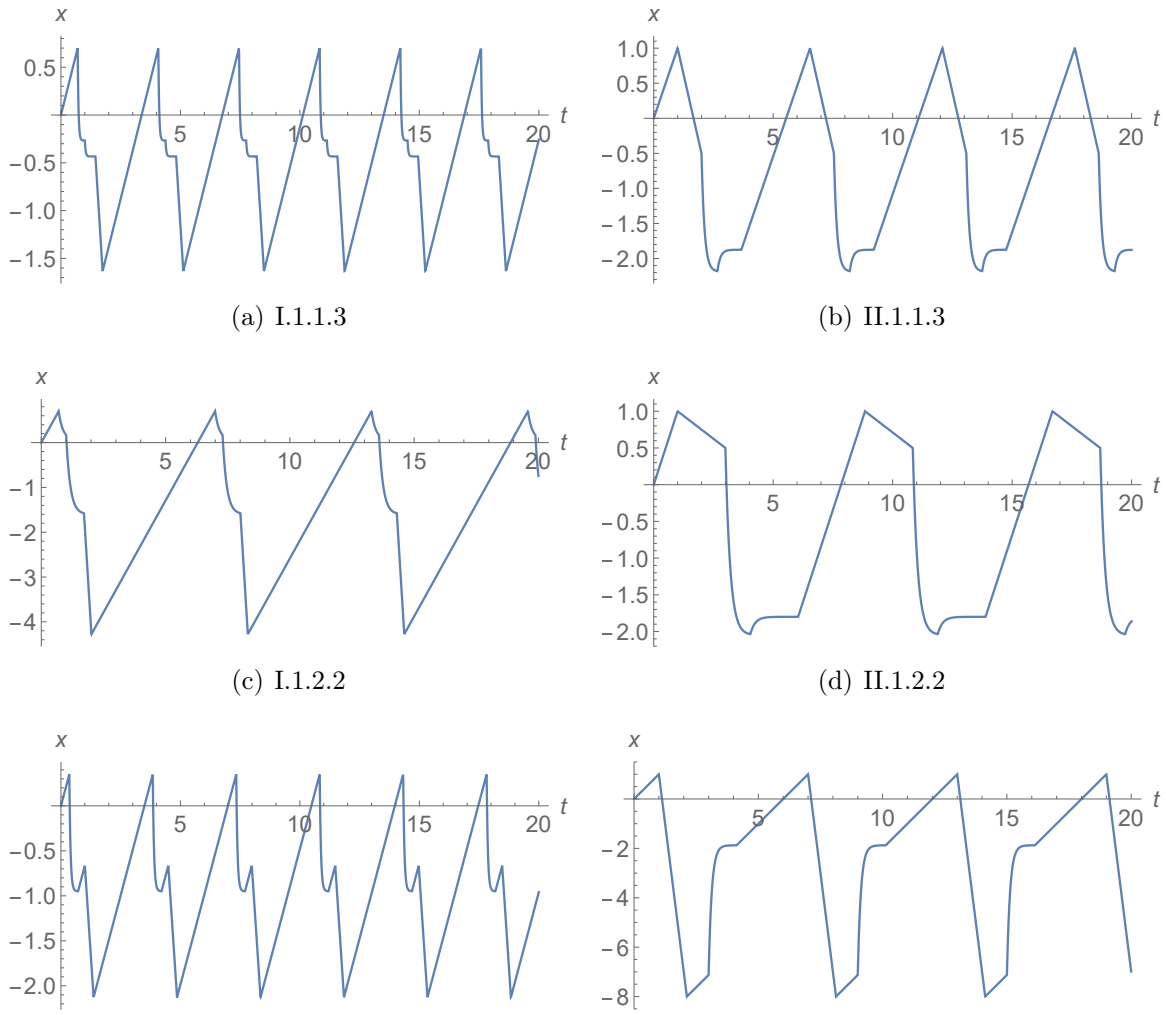
II.2.3 $\Delta=3,\ a=8,\ b=8,\ c=-2.$

Графики соответствующих решений релейного уравнения (7) изображены на рисунке 5.

Таким образом, получаем, что справедлива следующая

Лемма 1. *В каждом из приведенных выше случаев выбора параметров a,b,c,Δ решение релейного уравнения (5) с начальной функцией (6) совпадает с одной и той же кусочно линейной T_*-периодической функцией x_*, имеющей один нуль t_* на интервале $(0,T_*)$. Причем $x_*(t)>0$ при $t\in(0,t_*)$ и $x_*(t)<0$ при $t\in(t_*,T_*)$.*

Во всех случаях ограничений на параметры на каждом участке между точками переключения решение $x_\varphi(t)$ определяется однозначно из соответствующей задачи (A), (B), (C) или (D) с начальным условием $x\big|_{t=\widetilde{t}}=\widetilde{x}$, где $\widetilde{t}$ — это начало очередного промежутка, а начальное значение $\widetilde{x}$ выбирается из соображений непрерывности решения. Ниже приведены формулы для функции x_* в каждом случае выбора параметров a,b,c,Δ.



(a) I.1.1.3 (b) II.1.1.3

(c) I.1.2.2 (d) II.1.2.2

(e) I.2.3 (f) II.2.3

Рис. 5. Графики решений релейного уравнения в разных случаях выбора параметров a, b, c, Δ

Fig. 5. The graphs of solutions of the relay equation in different cases of the choice of the parameters a, b, c, Δ

I.1.1.3

$$x_*(t) = \begin{cases} x_A(0,\ 0;\ t) & \text{при } t \in [0, \Delta], \\ x_{C_1}(t) := x_C(\Delta,\ \Delta;\ t) & \text{при } t \in [\Delta, 1], \\ x_{D_1}(t) := x_D(1,\ x_{C_1}(1);\ t) & \text{при } t \in [1, t_* + \Delta], \\ x_{B_1}(t) := x_B\big(t_* + \Delta,\ x_{D_1}(t_* + \Delta);\ t\big) & \text{при } t \in [t_* + \Delta, t_* + 1], \\ x_{A_1}(t) := x_A\big(t_* + 1,\ x_{B_1}(t_* + 1);\ t\big) & \text{при } t \in [t_* + 1, T_*], \end{cases} \tag{15}$$

$$x_*(t + T_*) \equiv x_*(t),$$

где

$$t_* = -\frac{1}{b} \ln \frac{1/b + c}{1/b + c - \Delta} + \Delta,\ T_* = t_* + 1 - x_{B_1}(t_* + 1). \tag{16}$$

I.1.2.2 Для $x_*(t)$ и T_* справедливы те же формулы, что в случае I.1.1.3, но при

$$t_* = -\frac{1}{b} \ln \left(\frac{a/b - c}{x_{C_1}(1) + a/b - c} \right) + 1.$$

I.2.3

$$x_*(t) = \begin{cases} x_A(0,\ 0;\ t) & \text{при } t \in [0, \Delta], \\ x_{C_1}(t) := x_C(\Delta,\ \Delta;\ t) & \text{при } t \in [\Delta, t_* + \Delta], \\ x_{A_2}(t) := x_A(t_* + \Delta,\ x_{C_1}(t_* + \Delta);\ t) & \text{при } t \in [t_* + \Delta, 1], \\ x_{B_2}(t) := x_B\big(1,\ x_{A_2}(1);\ t\big) & \text{при } t \in [1, t_* + 1], \\ x_A\big(t_* + 1,\ x_{B_2}(t_* + 1);\ t\big) & \text{при } t \in [t_* + 1, T_*], \end{cases}$$

$$x_*(t + T_*) \equiv x_*(t),$$

где

$$t_* = -\frac{1}{b} \ln \left(\frac{-1/b - c}{\Delta - 1/b - c} \right) + \Delta,\ T_* = t_* + 1 - x_{B_2}(t_* + 1).$$

II.1.1.3

$$x_*(t) = \begin{cases} x_A(0,\ 0;\ t) & \text{при } t \in [0, 1], \\ x_{B_3}(t) := x_B(1,\ 1;\ t) & \text{при } t \in [1, \Delta], \\ x_{D_2}(t) := x_D(\Delta,\ x_{B_3}(\Delta);\ t) & \text{при } t \in [\Delta, t_* + 1], \\ x_{C_2}(t) := x_C\big(t_* + 1,\ x_{D_2}(t_* + 1);\ t\big) & \text{при } t \in [t_* + 1, t_* + \Delta], \\ x_A\big(t_* + \Delta,\ x_{C_2}(t_* + \Delta);\ t\big) & \text{при } t \in [t_* + \Delta, T_*], \end{cases}$$

$$x_*(t + T_*) \equiv x_*(t),$$

где

$$t_* = 1 + 1/a,\ T_* = t_* + \Delta - x_{C_2}(t_* + \Delta).$$

II.1.2.2 Для $x_*(t)$ и T_* справедливы те же формулы, что в случае II.1.1.3, но при

$$t_* = -\frac{1}{b} \ln \left(\frac{a/b - c}{x_{B_3}(\Delta) + a/b - c} \right) + 1.$$

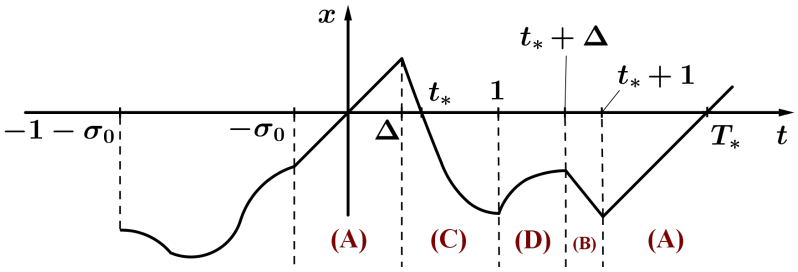


Рис. 6. Функция x_* в случае I.1.1.3
Fig. 6. The function x_* in the case of I.1.1.3

II.2.3

$$x_*(t) = \begin{cases} x_A(0,\ 0;\ t) & \text{при } t \in [0,1], \\ x_{B_3}(t) := x_B(1,\ 1;\ t) & \text{при } t \in [1, t_* + 1], \\ x_{A_3}(t) := x_A(t_* + 1,\ x_{B_3}(t_* + 1);\ t) & \text{при } t \in [t_* + 1, \Delta], \\ x_{C_3}(t) := x_C\big(\Delta,\ x_{A_3}(\Delta);\ t\big) & \text{при } t \in [\Delta, t_* + \Delta], \\ x_A\big(t_* + \Delta,\ x_{C_3}(t_* + \Delta);\ t\big) & \text{при } t \in [t_* + \Delta, T_*], \end{cases}$$

$$x_*(t + T_*) \equiv x_*(t),$$

где

$$t_* = 1 + 1/a,\ T_* = t_* + \Delta - x_{C_3}(t_* + \Delta).$$

В частности в случае I.1.1.3 явные формулы для решения $x_*(t)$ имеют вид

$$x_A(0,\ 0;\ t) \equiv t, \tag{17}$$

$$x_{C_1}(t) \equiv (\Delta - 1/b - c)\exp\big(-b(t - \Delta)\big) + 1/b + c, \tag{18}$$

$$x_{D_1}(t) \equiv (x_{C_1}(1) + a/b - c)\exp\big(-b(t - 1)\big) - a/b + c, \tag{19}$$

$$x_{B_1}(t) \equiv -a(t - t_* - \Delta) + x_{D_1}(t_* + \Delta), \tag{20}$$

$$x_{A_1}(t) \equiv t - t_* - 1 - a(1 - \Delta) + x_{D_1}(t_* + \Delta). \tag{21}$$

Схематичный график функции x_* изображен на рисунке 6.

Теперь наложим ограничения на до сих пор не выбранный параметр σ_0 из (8), (13). Будем считать выполненным условие

$$\sigma_0 < \min\{T_* - t_* - 1, T_* - t_* - \Delta\}, \tag{22}$$

которое обеспечивает принадлежность функции $x_\varphi(t + T_*)$ множеству (8) (или (13)).

Проделанный выбор параметров позволяет сформулировать утверждение о существовании и устойчивости периодического решения уравнения (5).

Теорема 1. *При выполнении сформулированных ограничений на параметры a, b, c, Δ и при всех достаточно малых $\varepsilon > 0$ уравнение (5) обладает единственным орбитально экспоненциально устойчивым циклом $x_*(t, \varepsilon)$, $x_*(-\sigma_0, \varepsilon) \equiv -\sigma_0$ периода $T_*(\varepsilon)$, который удовлетворяет предельным равенствам*

$$\lim_{\varepsilon \to 0} \max_{0 \leqslant t \leqslant T_*(\varepsilon)} |x_*(t, \varepsilon) - x_*(t)| = 0, \quad \lim_{\varepsilon \to 0} T_*(\varepsilon) = T_*. \tag{23}$$

3. Доказательство теоремы 1

Общая схема доказательства дается в статьях [1, 13, 14]. Для ее изложения введем некоторые обозначения. Зафиксируем произвольную функцию $\varphi(t) \in S(\sigma_0, q_1, q_2)$ и рассмотрим решение $x = x_\varphi(t, \varepsilon)$, $t \geqslant \sigma_0$, уравнения (5) с начальным условием $\varphi(t)$, $-1 - \sigma_0 \leqslant t \leqslant -\sigma_0$. Предположим, что уравнение $x_\varphi(t - \sigma_0, \varepsilon) = -\sigma_0$ имеет не менее 2-х положительных корней и обозначим 2-й корень через $t = T_\varphi$. Кроме того, зададим оператор последований Пуанкаре $\Pi_\varepsilon : C[-1 - \sigma_0, -\sigma_0] \to C[-1 - \sigma_0, -\sigma_0]$, определяя его равенством

$$\Pi_\varepsilon(\varphi) := x_\varphi(t + T_\varphi, \varepsilon),\ -1 - \sigma_0 \leqslant t \leqslant -\sigma_0 \text{ (или } -\Delta - \sigma_0 \leqslant t \leqslant -\sigma_0). \tag{24}$$

Дальнейший план доказательства такой же, как в упомянутых статьях. Для его реализации устанавливаются равномерные по φ и t асимптотические равенства для $x_\varphi(t, \varepsilon)$ на различных промежутках изменения t. Из них следует, что при подходящем выборе параметров $\sigma_0,\ q_1,\ q_2$ оператор Π_ε определен на множестве $S(\sigma_0, q_1, q_2)$ и преобразует его в себя, что позволяет использовать известный принцип Шаудера. Затем выполняется оценка нормы производной Фреше оператора Π_ε, из которой следует, что он является сжимающим на множестве $S(\sigma_0, q_1, q_2)$ и, тем самым, неподвижная точка в этом множестве единственна.

3.1. Построение асимптотики решения

В качестве примера построим асимптотику в случае I.1.1.3.

Построение асимптотики решения будет выполняться шагами по промежуткам изменения t.

Рассмотрим сперва отрезок $t \in [-\sigma_0, \Delta - \sigma_0]$. Считаем, что наряду с ограничением (22) выполняется условие $\sigma_0 < \Delta$. Здесь $t - 1$ и $t - \Delta$ принадлежат отрезкам, вложенным в $[-1-\sigma_0, -\sigma_0]$, где функции $x(t-1, \varepsilon)$, $x(t-\Delta, \varepsilon)$ совпадают с функциями $\varphi(t-1)$, $\varphi(t-\Delta)$. Согласно (8), это означает, что $x(t-1, \varepsilon) < -q_2$, $x(t-\Delta, \varepsilon) < -q_2$, следовательно, учитывая условия (2), получаем, что

$$F(x(t-1, \varepsilon), \varepsilon) = 1 + O\big(\exp(-q_2/\varepsilon)\big), \tag{25}$$

$$G\big(x(t-\Delta, \varepsilon), \varepsilon\big) = O\big(\exp(-q_2/\varepsilon)\big).$$

Таким образом, на рассматриваемом отрезке имеем дело с задачей Коши

$$\dot{x} = 1 + O\big(\exp(-q_2/\varepsilon)\big),\ x\big|_{t=-\sigma_0} = -\sigma_0,$$

откуда приходим к равномерному по φ асимптотическому равенству

$$x_\varphi(t, \varepsilon) = t + O\big(\exp(-q/\varepsilon)\big) \text{ при } t \in [-\sigma_0, \Delta - \sigma_0]. \tag{26}$$

Здесь и далее q обозначает подходящую положительную константу, точное значение которой не важно.

Для дальнейшего построения асимптотики решения $x_\varphi(t, \varepsilon)$ уравнения (5) зафиксируем параметр

$$\alpha \in (1/2, 1).$$

Рассматривая отрезок $[\Delta-\sigma_0, \Delta-\varepsilon^\alpha]$, получаем аналогичную предыдущей задачу Коши, но с остатками порядка $O\big(\exp(-q\varepsilon^{\alpha-1})\big)$ в нелинейности:

$$\dot{x} = 1 + O\big(\exp(-q\varepsilon^{\alpha-1})\big),\ x\big|_{t=\Delta-\sigma_0} = \Delta - \sigma_0 + O\big(\exp(-q/\varepsilon)\big).$$

Таким образом, на указанном промежутке для решения справедлива формула

$$x_\varphi(t,\varepsilon) = t + O\big(\exp(-q/\varepsilon)\big) \text{ при } t \in [\Delta - \sigma_0, \Delta - \varepsilon^\alpha]. \tag{27}$$

Теперь рассмотрим отрезок $[\Delta-\varepsilon^\alpha, \Delta+\varepsilon^\alpha]$, на котором решение релейного уравнения терпит излом. На этом участке для функции $F\big(x(t-1,\varepsilon),\varepsilon\big)$ сохраняется формула (25), а функция $G\big(x(t-\Delta,\varepsilon),\varepsilon\big)$ с учетом (26) принимает вид:

$$G\big(x(t-\Delta,\varepsilon),\varepsilon\big) = g\left(\exp\frac{t-\Delta+O\big(\exp(-q/\varepsilon)\big)}{\varepsilon}\right).$$

На текущем участке имеем дело с задачей Коши

$$\begin{cases} \dot{x} = 1 + b(c-x)\cdot g\left(\exp\dfrac{t-\Delta+O\big(\exp(-q/\varepsilon)\big)}{\varepsilon}\right) + O\big(\exp(-q_2/\varepsilon)\big), \\ x\big|_{t=\Delta-\varepsilon^\alpha} = \Delta - \varepsilon^\alpha + O\big(\exp(-q/\varepsilon)\big). \end{cases} \tag{28}$$

Решение задачи (28) будем искать в виде

$$x_\varphi(t,\varepsilon) = \Delta + \varepsilon w_1(\tau)\big|_{\tau=(t-\Delta)/\varepsilon} + \delta(t,\varepsilon), \tag{29}$$

где функция $w_1(\tau)$ задается равенством

$$w_1(\tau) := \tau + b(c-\Delta)\int\limits_{-\infty}^{\tau} g(\exp s)ds, \tag{30}$$

а $\delta(t,\varepsilon)$ — подлежащий определению остаток.

Докажем, что остаток $\delta(t,\varepsilon)$ является равномерно по φ и t экспоненциально малым. С этой целью сперва выясним асимптотическое поведение функции w_1.

Лемма 2. *Для функции w_1 справедливы следующие асимптотические равенства:*

$$w_1(\tau) = \tau + O(\exp\tau) \text{ при } \tau \to -\infty, \tag{31}$$

$$w_1(\tau) = (1 + bc - b\Delta)\tau + d_1 b(c-\Delta) + O\big(\exp(-\tau)\big) \text{ при } \tau \to +\infty, \text{ где} \tag{32}$$

$$d_1 := \int\limits_0^1 \frac{g(u)}{u}du + \int\limits_1^{+\infty}\frac{g(u)-1}{u}du. \tag{33}$$

Отметим, что в равенстве (32) коэффициент $1+bc-b\Delta$ при τ совпадает с угловым коэффициентом касательной к функции $x_*(t)$ в точке Δ справа.

Равенство (31) вытекает непосредственно из свойств (2) функции g.

Для доказательства равенства (32) представим интеграл (30) в виде суммы:

$$\int\limits_{-\infty}^{\tau} g(\exp s)ds = \int\limits_{-\infty}^{0} g(\exp s)ds + \int\limits_{0}^{\tau} g(\exp s)ds.$$

Первый интеграл после замены $u = \exp s$ преобразуется в $\int\limits_0^1 \frac{g(u)}{u}du$, который в силу (2) равен конечному числу. Для второго, учитывая свойства (2), верно

$$\int\limits_{0}^{\tau} g(\exp s)ds = \tau + \int\limits_{0}^{+\infty} \big(g(\exp s) - 1\big)ds - \int\limits_{\tau}^{+\infty} \big(g(\exp s) - 1\big)ds =$$

$$= \tau + \int\limits_{1}^{+\infty} \frac{g(u) - 1}{u}du + O\big(\exp(-\tau)\big),$$

где интеграл $\int\limits_1^{+\infty} \frac{g(u)-1}{u}du$ сходится в силу свойств (2) функции g. Таким образом, лемма 2 доказана.

Далее, подставляя (29), (30) в (28) и учитывая асимптотические свойства функции w_1 из леммы 2, получаем задачу Коши для остатка δ:

$$\dot\delta = b(c-\Delta)\cdot\left(g\left(\exp\frac{t-\Delta+O\big(\exp(-q/\varepsilon)\big)}{\varepsilon}\right) - g\left(\exp\frac{t-\Delta}{\varepsilon}\right)\right) - \Big(b(t-\Delta)+$$

$$+\varepsilon b^2(c-\Delta)\int\limits_{-\infty}^{(t-\Delta)/\varepsilon} g(\exp s)ds + \delta(t,\varepsilon)\Big)\cdot g\Big(\exp\frac{t-\Delta+O(\exp(-q/\varepsilon))}{\varepsilon}\Big), \quad (34)$$

$$\delta\big|_{t=\Delta-\varepsilon^\alpha} = -\varepsilon b(c-\Delta)\int\limits_{-\infty}^{-\varepsilon^{\alpha-1}} g(\exp s)ds + O\big(\exp(-q/\varepsilon)\big). \quad (35)$$

Принимая во внимание неравенство

$$|g(x_1) - g(x_2)| \leqslant \frac{M_1}{1+\min\big(x_1^2, x_2^2\big)}|x_1 - x_2| \text{ при всех } x_1, x_2 \in \mathbb{R}_+, \quad (36)$$

асимптотические свойства интеграла $\int\limits_{-\infty}^{\tau} g(\exp s)ds$ и то, что t изменяется на отрезке длины, пропорциональной ε^α, получаем асимптотику правых частей равенств (34) и (35):

$$\begin{cases} \dot\delta = -\delta\cdot g\left(\exp\dfrac{t-\Delta+O\big(\exp(-q/\varepsilon)\big)}{\varepsilon}\right) + O(\varepsilon^\alpha), \\ \delta\big|_{t=\Delta-\varepsilon^\alpha} = O\big(\varepsilon\exp(-q\varepsilon^{\alpha-1})\big). \end{cases} \quad (37)$$

Из вида задачи Коши (37) следует, что $\delta(t,\varepsilon) = O\big(\varepsilon \exp(-q\varepsilon^{\alpha-1})\big)$ равномерно по φ.

Дальнейшее построение асимптотики решения $x_\varphi(t,\varepsilon)$ происходит аналогичным образом, поэтому ограничимся лишь результатом. Перед его формулировкой введем обозначение для константы

$$d_2 := \int\limits_0^1 \frac{f(u)-1}{u}du + \int\limits_1^{+\infty} \frac{f(u)+a}{u}du.$$

Отметим, что в силу свойств (2) функции f соответствующие несобственные интегралы сходятся. Кроме того, определим функции w_2, w_3, w_4, аналогичные (30):

$$w_2(\tau) := \dot{x}_{C_1}(1)\tau + bd_1(c-\Delta) + \int\limits_{-\infty}^{\tau} \big(f(\exp s) - 1\big)ds, \tag{38}$$

$$w_3(\tau) := \dot{x}_{D_1}\big(t_\varphi(\varepsilon)+\Delta\big)\tau + bd_1(c-\Delta) + d_2 + \\ + b\Big(c - x_{D_1}\big(t_\varphi(\varepsilon)+\Delta\big)\Big)\int\limits_{-\infty}^{\tau}\Big(g\Big(\exp\big((1+bc)s\big)\Big) - 1\Big)ds, \tag{39}$$

$$w_4(\tau) := -a\tau + bd_1(c-\Delta) + d_2 - \frac{d_1}{1+bc} + \int\limits_{-\infty}^{\tau}\Big(f\Big(\exp\big((1+bc)s\big)\Big) + a\Big)ds. \tag{40}$$

Напомним, что d_1 определяется формулой (33). Сформулируем утверждение, описывающее асимптотическое поведение введенных функций (38), (39), (40) при $\tau \to -\infty$ и $\tau \to +\infty$.

Лемма 3. *Для функций w_2, w_3, w_4 справедливы следующие асимптотические равенства:*

$$w_2(\tau) = \dot{x}_{C_1}(1)\tau + bd_1(c-\Delta) + O\big(\exp(\tau)\big) \text{ при } \tau \to -\infty,$$

$$w_2(\tau) = \dot{x}_{D_1}(1)\tau + bd_1(c-\Delta) + d_2 + O\big(\exp(-\tau)\big) \text{ при } \tau \to +\infty;$$

$$w_3(\tau) = \dot{x}_{D_1}(t_\varphi(\varepsilon)+\Delta)\tau + bd_1(c-\Delta) + d_2 + O\Big(\exp\big(-(1+bc)\tau\big)\Big) \text{ при } \tau \to -\infty,$$

$$w_3(\tau) = -a\tau + bd_1(c-\Delta) + d_2 - \frac{d_1}{1+bc} + O\Big(\exp\big((1+bc)\tau\big)\Big) \text{ при } \tau \to +\infty;$$

$$w_4(\tau) = -a\tau + bd_1(c-\Delta) + d_2 - \frac{d_1}{1+bc} + O\Big(\exp\big(-(1+bc)\tau\big)\Big) \text{ при } \tau \to -\infty,$$

$$w_4(\tau) = \tau + bd_1(c-\Delta) + d_2 - \frac{d_1+d_2}{1+bc} + O\Big(\exp\big((1+bc)\tau\big)\Big) \text{ при } \tau \to +\infty.$$

Конструктивно, методом шагов, а также с использованием леммы 3, устанавливается справедливость следующего утверждения.

Лемма 4. *Уравнение (5) с произвольной начальной функцией φ из класса (8) имеет решение $x_\varphi(t,\varepsilon)$ с асимптотикой*

1. $x_\varphi(t,\varepsilon) = t + O\big(\exp(-q/\varepsilon)\big)$ *при* $t \in [-\sigma_0, \Delta - \sigma_0]$;

2. $x_\varphi(t,\varepsilon) = t + O\big(\exp(-q/\varepsilon)\big)$ *при* $t \in [\Delta - \sigma_0, \Delta - \varepsilon^\alpha]$;

3. $x_\varphi(t,\varepsilon) = \Delta + \varepsilon w_1(\tau)\big|_{\tau=(t-\Delta)/\varepsilon} + O\big(\varepsilon\exp(-q\varepsilon^{\alpha-1})\big)$ *при* $t \in [\Delta - \varepsilon^\alpha, \Delta + \varepsilon^\alpha]$;

4. $x_\varphi(t,\varepsilon) = x_{C_1}(t) + \varepsilon b d_1(c-\Delta) + O(\varepsilon^{2\alpha})$ *при* $t \in [\Delta + \varepsilon^\alpha, 1 - \varepsilon^\alpha]$;

5. $x_\varphi(t,\varepsilon) = x_{C_1}(1) + \varepsilon w_2(\tau)\big|_{\tau=(t-1)/\varepsilon} + O(\varepsilon^{2\alpha})$ *при* $t \in [1 - \varepsilon^\alpha, 1 + \varepsilon^\alpha]$;

6. $x_\varphi(t,\varepsilon) = x_{D_1}(t) + \varepsilon\big(bd_1(c-\Delta) + d_2\big) + O(\varepsilon^{2\alpha})$ *при* $t \in [1 + \varepsilon^\alpha, t_\varphi(\varepsilon) + \Delta - \varepsilon^\alpha]$, *где* $t_\varphi(\varepsilon)$ — *первый положительный корень уравнения* $x_\varphi(t,\varepsilon) = 0$, *который лежит при этом на интервале* $(\Delta + \varepsilon^\alpha, 1 - \varepsilon^\alpha)$ *(см. п. 4)*;

7. $x_\varphi(t,\varepsilon) = x_{D_1}(t_\varphi(\varepsilon) + \Delta) + \varepsilon w_3(\tau)\big|_{\tau=(t-t_\varphi(\varepsilon)-\Delta)/\varepsilon} + O(\varepsilon^{2\alpha})$ *при* $t \in [t_\varphi(\varepsilon) + \Delta - \varepsilon^\alpha, t_\varphi(\varepsilon) + \Delta + \varepsilon^\alpha]$;

8. $x_\varphi(t,\varepsilon) = x_{B_1}(t) + \varepsilon\Big(bd_1(c-\Delta) + d_2 - \dfrac{d_1}{1+bc}\Big) + O(\varepsilon^{2\alpha})$ *при* $t \in [t_\varphi(\varepsilon) + \Delta + \varepsilon^\alpha, t_\varphi(\varepsilon) + 1 - \varepsilon^\alpha]$;

9. $x_\varphi(t,\varepsilon) = x_{B_1}(t_\varphi(\varepsilon) + 1) + \varepsilon w_4(\tau)\big|_{\tau=(t-t_\varphi(\varepsilon)-1)/\varepsilon} + O(\varepsilon^{2\alpha})$ *при* $t \in [t_\varphi(\varepsilon) + 1 - \varepsilon^\alpha, t_\varphi(\varepsilon) + 1 + \varepsilon^\alpha]$;

10. $x_\varphi(t,\varepsilon) = x_{A_1}(t) + \varepsilon\Big(bd_1(c-\Delta) + d_2 - \dfrac{d_1+d_2}{1+bc}\Big) + O(\varepsilon^{2\alpha})$ *при* $t \in [t_\varphi(\varepsilon) + 1 + \varepsilon^\alpha, T_\varphi(\varepsilon) - \sigma_0/2]$, *где* $T_\varphi(\varepsilon)$ — *второй положительный корень уравнения* $x_\varphi(t - \sigma_0, \varepsilon) = -\sigma_0$, *для которого справедлива формула*

$$T_\varphi(\varepsilon) = t_\varphi(\varepsilon) + 1 - x_{B_1}\big(t_\varphi(\varepsilon)\big) + O(\varepsilon). \tag{41}$$

Все остатки здесь равномерны по $\varphi \in S(\sigma_0, q_1, q_2)$ *и* t *из соответствующих промежутков.*

Тем самым, построена асимптотика решения $x_\varphi(t,\varepsilon)$ с начальной функцией $\varphi \in S(\sigma_0, q_1, q_2)$ (см. (8)). Причем из формул леммы 4 следует, что для решения $x_\varphi(t,\varepsilon)$ и корня $T_\varphi(\varepsilon)$ верны равномерные по $\varphi \in S(\sigma_0, q_1, q_2)$ оценки

$$\max_{t\in[-\sigma_0, T_* - \sigma_0/2]} |x_\varphi(t,\varepsilon) - x_*(t)| = O(\varepsilon), \quad T_\varphi(\varepsilon) = T_* + O(\varepsilon), \tag{42}$$

где $x_*(t)$, напомним, — функция (15), а T_* — период этой функции, задаваемый формулой (16).

3.2. Существование, единственность и устойчивость периодического решения

Следующий шаг наших рассуждений состоит в доказательстве существования, единственности и устойчивости периодического решения.

Из равенств (42) заключаем, что оператор (24) определен на множестве $S(\sigma_0, q_1, q_2)$, причем равномерно по φ

$$\max_{t\in[-1-\sigma_0,-\sigma_0]} |x_\varphi(t+T_\varphi,\varepsilon) - x_*(t)| = O(\varepsilon). \tag{43}$$

Для доказательства включения

$$\Pi_\varepsilon\big(S(\sigma_0, q_1, q_2)\big) \subset S(\sigma_0, q_1, q_2) \tag{44}$$

так же, как в статьях [1, 13, 14], наложим на параметры q_1, q_2 ограничения

$$q_1 > -\min_{t\in[-1-\sigma_0,-\sigma_0]} x_*(t),\ 0 < q_2 < -\max_{t\in[-1-\sigma_0,-\sigma_0]} x_*(t) \tag{45}$$

и рассмотрим множество функций $\hat{S}(\sigma_0, q_1, q_2)$, получающееся из $S(\sigma_0, q_1, q_2)$ при замене в (8) нестрогих равенств строгими. В силу (43) требуемое включение (44) будет выполняться при всех достаточно малых $\varepsilon > 0$ при условии

$$x_*(t) \in \hat{S}(\sigma_0, q_1, q_2), \tag{46}$$

что верно в силу (45).

Таким образом, оператор Π_ε является компактным и преобразует в себя замкнутое ограниченное и выпуклое множество $S(\sigma_0, q_1, q_2)$. Отсюда, используя принцип Шаудера, получаем, что этот оператор имеет в $S(\sigma_0, q_1, q_2)$ по крайней мере одну неподвижную точку $\varphi = \varphi_*(t,\varepsilon)$. А значит, решение $x_*(t,\varepsilon)$ уравнения (5) с начальной функцией $\varphi_*(t,\varepsilon)$, $-1-\sigma_0 \leqslant t \leqslant -\sigma_0$, является периодическим с периодом $T_*(\varepsilon) = T_\varphi\big|_{\varphi=\varphi_*}$ и в силу (42), (43) удовлетворяет свойствам (23).

Единственность и устойчивость периодического решения $x_*(t,\varepsilon)$ доказываются стандартным образом на основе оценки производной Фреше оператора Π_ε, получаемой исходя из вида решения $x_\varphi(t,\varepsilon)$ (см. [1, 13, 14]).

4. Заключение

Напомним, что поскольку решение системы (3) отыскивается в виде бегущей волны (4), то на период искомого решения накладывается ограничение: $T = m\Delta/k$, где m — число уравнений в системе (1), $k \in \mathbb{N}$. В связи с этим, далее, чтобы подчеркнуть зависимость периодов функций $x_*(t)$, $x_*(t,\varepsilon)$ (см. (15) и (23)) от фазового сдвига Δ, будем обозначать их $T_*(\Delta)$, $T_*(\Delta,\varepsilon)$ соответственно. Отметим, что из формул (16)–(20) следует справедливость равенства

$$T_*(\Delta) = -\frac{1}{b}\ln\frac{1/b+c}{1/b+c-\Delta} + 1 + a(-a\Delta + a/b - c + (1/b+c)\exp(-b\Delta) - \\ - \frac{(1/b+c)(1+a)}{1+bc-b\Delta}\exp\big(-b(2\Delta-1)\big). \tag{47}$$

В силу (23) верно представление $T_*(\varepsilon,\Delta) = T_*(\Delta) + O(\varepsilon)$. Потребуем, чтобы $T_*(\Delta) + O(\varepsilon) = m\Delta/k$, где $T_*(\Delta)$ описывается формулой (47), m определяет число нейронов в модели (1), $k \in \mathbb{N}$.

Подводя итог, отметим, что настоящая работа дополняет исследование, проделанное в работе [1]. Доказано, что для уравнения (5) можно выделить ровно шесть случаев значений параметров a, b, c, Δ, при которых существует устойчивое периодическое решение с одним промежутком положительности и одним промежутком отрицательности на периоде (см. рис. 2), причем длина отрицательной фазы не меньше наибольшего из запаздываний уравнения (5). Таким образом, полностью изучен вопрос существования и устойчивости периодических решений указанного вида для уравнения (5) в широкой области параметров.

Список литературы / References

[1] Глызин С. Д., Колесов А.Ю, Розов Н. Х., "Об одном способе математического моделирования химических синапсов", *Дифференциальные уравнения*, **49**:10 (2013), 1227–1244; [Glyzin S. D., Kolesov A. Yu., Rozov N. Kh., "On a Method for Mathematical Modeling of Chemical Synapses", *Differential Equations*, **49**:10 (2013), 1193–1210].

[2] Somers D., Kopell N., "Rapid synchronization through fast threshold modulation", *Biol. Cybern.*, **68** (1993), 393–407.

[3] Somers D., Kopell N., "Anti-phase solutions in relaxation oscillators coupled through excitatory interactions", *J. Math. Biol.*, **33** (1995), 261–280.

[4] Izhikevich E. M., *Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting*, MIT Press, 2010.

[5] FitzHugh R. A., "Impulses and physiological states in theoretical models of nerve membrane", *Biophysical J.*, **1** (1961), 445–466.

[6] Terman D., "An Introduction to Dynamical Systems and Neuronal Dynamics", *Tutorials in Mathematical Biosciences I, Lecture Notes in Mathematics*, **1860** (2005), 21–68.

[7] Hutchinson G. E., "Circular causal systems in ecology", *Ann. N. Y. Acad. of Sci.*, **50** (1948), 221–246.

[8] Колесов А.Ю, Мищенко Е. Ф., Розов Н. Х., "Реле с запаздыванием и его C^1-аппроксимация", *Тр. Мат. ин-та им. В.А. Стеклова РАН*, **216** (1997), 126–153; [Kolesov A. Yu., Mishchenko E. F., Rozov N. Kh., "Relay with delay and its C^1-approximation", *Proceedings of the Steklov Institute of Mathematics*, **216** (1997), 119–146].

[9] Глызин С. Д., Колесов А.Ю, Розов Н. Х., "Релаксационные автоколебания в нейронных системах. I", *Дифференциальные уравнения*, **47**:7 (2011), 919–932; [Glyzin S. D., Kolesov A. Yu., Rozov N. Kh., "Relaxation self-oscillations in neuron systems: I", *Differential Equations*, **47**:7 (2011), 927–941].

[10] Глызин С. Д., Колесов А.Ю, Розов Н. Х., "Релаксационные автоколебания в нейронных системах. II", *Дифференциальные уравнения*, **47**:12 (2011), 1675–1692; [Glyzin S. D., Kolesov A. Yu., Rozov N. Kh., "Relaxation self-oscillations in neuron systems: II", *Differential Equations*, **47**:12 (2011), 1697–1713].

[11] Глызин С. Д., Колесов А.Ю, Розов Н. Х., "Релаксационные автоколебания в нейронных системах. III", *Дифференц. уравнения*, **48**:2 (2012), 155–170; [Glyzin S. D., Kolesov A. Yu., Rozov N. Kh., "Relaxation self-oscillations in neuron systems: III", *Differential Equations*, **48**:2 (2012), 159–175].

[12] Глызин С. Д., Колесов А.Ю, Розов Н. Х., "Дискретные автоволны в нейронных системах", *ЖВМ и МФ*, **52**:5 (2012), 840–858; [Glyzin S. D., Kolesov A. Yu., Rozov N. Kh., "Discrete autowaves in neural systems", *Computational Mathematics and Mathematical Physics*, **52**:5 (2012), 702–719].

[13] Колесов А.Ю, Мищенко Е. Ф., Розов Н. Х., "Об одной модификации уравнения Хатчинсона", *ЖВМ и МФ*, **50**:12 (2010), 2099–2112; [Kolesov A. Yu., Mishchenko E. F.,

Rozov N. Kh., "A modification of Hutchinson's equation", *Computational Mathematics and Mathematical Physics*, **50**:12 (2010), 1990–2002].

[14] Преображенская М. М., "Существование и устойчивость релаксационных циклов в нейродинамической модели с двумя запаздываниями", *Вестник НИЯУ МИФИ*, **5**:4 (2016), 351–366; [Preobrazhenskaia M. M., "Existence and stability of relaxation cycles in a neurodynamic model with two delays", *Vestnik NIYaU MIFI*, **5**:4 (2016), 351–366].

[15] Глызин С. Д., Колесов А.Ю, Розов Н. Х., "Релаксационные автоколебания в сетях импульсных нейронов", *УМН*, **70**:3(423) (2015), 3–76; [Glyzin S. D., Kolesov A. Yu., Rozov N. Kh., "Self-excited relaxation oscillations in networks of impulse neurons", *Russian Math. Surveys*, **70**:3 (2015), 383–452].

[16] Глызин С. Д., Колесов А.Ю, Розов Н. Х., "Релаксационные автоколебания в сетях Хопфилда с запаздыванием", *Изв. РАН. Сер. матем.*, **77**:2 (2013), 53–96; [Glyzin S. D., Kolesov A. Yu., Rozov N. Kh., "Relaxation self-oscillations in Hopfield networks with delay", *Izvestiya: Mathematics*, **77**:2 (2013), 271–312].

Preobrazhenskaia M. M., "Relaxation Cycles in a Model of Synaptically Interacting Oscillators", *Modeling and Analysis of Information Systems*, **24**:2 (2017), 186–204.

DOI: 10.18255/1818-1015-2017-2-186-204

Abstract. In this paper the mathematical model of a neural network with a ring synaptic interaction elements is considered. The model is a system of scalar nonlinear differential-difference equations, the right parts of which depend on a large parameter. The unknown functions included in the system characterize the membrane potentials of the neurons. The search of relaxation cycles within the system of equations is interested. To this end solutions of the task are finded in the form of discrete traveling waves. It allows to research a scalar nonlinear differential-difference equations with two delays instead of system. Further, a limit a object that represents a relay equation with two delays is defined by large parameter tends to infinity. There are six cases of restrictions on the parameters. In every case exist alone periodic solution of relay equation started from initial function from suitable function class. It is structurally proved by using the step method. Next, the existence of a relaxation periodic solutions of a singularly perturbed equation with two delays is proved by using Poincare operator and Schauder principle. The asymptotics of this solution is constructed, and then it is proved that the solution is close to decision of the relay equation. Because of the exponential estimate Frechet derivative of the Poincare operator implies the uniqueness and stability of solutions of differential-difference equation with two delays.

Keywords: relaxation oscillations, delay, large parameter, synaptic connection

About the authors:
Margarita M. Preobrazhenskaia, orcid.org/0000-0002-7032-1155,
P.G. Demidov Yaroslavl State University, 14 Sovetskaya str., Yaroslavl 150003, Russia,
Scientific Center in Chernogolovka RAS, 9 Lesnaya str., Chernogolovka, Moscow region 142432, Russia,
e-mail: rita.preo@gmail.com

Acknowledgments:
This work was supported by the Russian Science Foundation (project nos. №14-21-00158).

Моделирование и анализ информационных систем. Т. 24, № 2 (2017), с. 205–214
Modeling and Analysis of Information Systems. Vol. 24, No 2 (2017), pp. 205–214

DOI: 10.18255/1818-1015-2017-2-205-214

УДК 512.71

О гипотезах Тэйта для дивизоров на расслоенном многообразии и его общем схемном слое в случае конечной характеристики

Прохорова Т. В.

получена 12 декабря 2016

Аннотация. В работе изучаются взаимоотношения между гипотезой Тэйта для дивизоров на расслоенном многообразии над конечным полем и гипотезой Тэйта для дивизоров на общем схемном слое при условии, что общий схемный слой имеет иррегулярность нуль. Пусть $\pi : X \to C$ – сюръективный морфизм гладких проективных многообразий над конечным полем $\mathbb{F}_q$ характеристики p, C – кривая, общий схемный слой морфизма π является гладким многообразием V над полем $k = \kappa(C)$ рациональных функций кривой C, $\overline{k}$ – алгебраическое замыкание поля k, k^s – его сепарабельное замыкание, $\mathrm{NS}(V)$ – группа Нерона – Севери классов дивизоров на многообразии V по модулю алгебраической эквивалентности, причем выполнены следующие условия: $H^1(V \otimes \overline{k}, \mathcal{O}_{V\otimes\overline{k}}) = 0$, $\mathrm{NS}(V) = \mathrm{NS}(V \otimes \overline{k})$. Если для простого числа l, не делящего $\mathrm{Card}([\mathrm{NS}(V)]_{\mathrm{tors}})$ и отличного от характеристики поля $\mathbb{F}_q$, верно соотношение $\mathrm{NS}(V) \otimes \mathbb{Q}_l \twoheadrightarrow [H^2(V \otimes k^{\mathrm{s}}, \mathbb{Q}_l(1))]^{\mathrm{Gal}(k^{\mathrm{s}}/k)}$ (другими словами, если верна гипотеза Тэйта для дивизоров на V), то для любого простого числа $l \neq \mathrm{char}(\mathbb{F}_q)$ гипотеза Тэйта верна для дивизоров на X: $\mathrm{NS}(X) \otimes \mathbb{Q}_l \twoheadrightarrow [H^2(X \otimes \overline{\mathbb{F}}_q, \mathbb{Q}_l(1))]^{\mathrm{Gal}(\overline{\mathbb{F}}_q/\mathbb{F}_q)}$. В частности, из этого результата следует гипотеза Тэйта для дивизоров на арифметической модели К3 – поверхности над достаточно большим глобальным полем конечной характеристики, отличной от 2.

Ключевые слова: гипотеза Тэйта, глобальное поле, группа Брауэра, арифметическая модель, К 3 – поверхность

Для цитирования: Прохорова Т. В., "О гипотезах Тэйта для дивизоров на расслоенном многообразии и его общем схемном слое в случае конечной характеристики", *Моделирование и анализ информационных систем*, **24**:2 (2017), 205–214.

Об авторах:
Прохорова Татьяна Вячеславовна, orcid.org/0000-0002-6883-2087, канд. физ.-мат. наук,
Владимирский государственный университет им. А.Г. и Н.Г. Столетовых,
ул. Горького, 87, г. Владимир, 600000 Россия,
e-mail: tvprokhorova@mail.ru

Введение

Пусть V – гладкое проективное многообразие над глобальным полем $k = \kappa(C)$ рациональных функций на гладкой проективной кривой C над конечным полем $\mathbb{F}_q$ характеристики p. Предположим, что существует проективный плоский $\mathbb{F}_q$-морфизм

$\pi : X \to C$, где X – регулярная схема, общий схемный слой π изоморфен V (мы называем $\pi : X \to C$ *арифметической моделью* V).

В работе доказано, что из гипотезы Тэйта для дивизоров на регулярном многообразии V над достаточно большим глобальным полем k следует гипотеза Тэйта для дивизоров на X.

Автор благодарит С. Г. Танкеева за ценные советы.

Основные результаты

Предложение 1. [1, предложение 0.3] *Пусть X – гладкое проективное многообразие над конечным полем $\mathbb{F}_q$. Если для простого числа $l \neq \mathrm{char}(\mathbb{F}_q)$ верна гипотеза Тэйта о дивизориальных циклах:*

$$\mathrm{NS}(X) \otimes_{\mathbb{Z}} \mathbb{Q}_l = H^2(X \otimes \overline{\mathbb{F}}_q, \mathbb{Q}_l(1))^{\mathrm{Gal}(\overline{\mathbb{F}}_q/\mathbb{F}_q)},$$

то это соотношение верно для всех $l \neq \mathrm{char}(\mathbb{F}_q)$.

Предложение 2. [2, предложение 4.3] *Пусть X – гладкое проективное многообразие над конечным полем $\mathbb{F}_q$. Следующие утверждения эквивалентны*:

a) $\mathrm{NS}(X) \otimes_{\mathbb{Z}} \mathbb{Q}_l = H^2(X \otimes \overline{\mathbb{F}}_q, \mathbb{Q}_l(1))^{\mathrm{Gal}(\overline{\mathbb{F}}_q/\mathbb{F}_q)}$;

b) группа $\mathrm{Br}'(X)(l)$ *конечна*;

c) каноническое отображение $\mathbb{Z}_l \otimes \mathrm{Pic}\, X \to H^2(X_{et}, \mathbb{Z}_l(1))$ *биективно* ;

d) порядок полюса дзета-функции Хассе – Вейля
$Z(X, t)$ *в точке* $t = q^{-1}$ *равен рангу группы* $\mathrm{Pic}\, X$.

В этой работе мы докажем следующий основной результат:

Теорема 1. *Пусть $\pi : X \to C$ – сюръективный морфизм гладких проективных многообразий над конечным полем $\mathbb{F}_q$ характеристики p, C – кривая, общий схемный слой морфизма π является гладким многообразием V над полем $k = \kappa(C)$, $H^1(V \otimes \overline{k}, \mathcal{O}_{V \otimes \overline{k}}) = 0$, $\mathrm{NS}(V) = \mathrm{NS}(V \otimes \overline{k})$. Если для простого числа l, не делящего $\mathrm{Card}([\mathrm{NS}(V)]_{\mathrm{tors}})$ и отличного от характеристики поля $\mathbb{F}_q$, верно соотношение*

$$\mathrm{NS}(V) \otimes \mathbb{Q}_l \twoheadrightarrow [H^2(V \otimes k^{\mathrm{s}}, \mathbb{Q}_l(1))]^{\mathrm{Gal}(k^{\mathrm{s}}/k)}$$

(другими словами, если верна гипотеза Тэйта для дивизоров на V), то для любого простого числа $l \neq \mathrm{char}(\mathbb{F}_q)$ гипотеза Тэйта верна для дивизоров на X:

$$\mathrm{NS}(X) \otimes \mathbb{Q}_l \twoheadrightarrow [H^2(X \otimes \overline{\mathbb{F}}_q, \mathbb{Q}_l(1))]^{\mathrm{Gal}(\overline{\mathbb{F}}_q/\mathbb{F}_q)}.$$

Доказательство. Будем обозначать через $\kappa(y)$ поле вычетов точки $y \in X$.

Пусть $i_y : \mathrm{Spec}\, \kappa(y) \to X$ – каноническое вложение $y \in X$ и

$$\mathrm{Div}_X^{\mathrm{vert}} = \bigoplus_{\substack{y \in X \setminus V \\ \mathrm{codim}_X(y) = 1}} i_{y*}\mathbb{Z}$$

– пучок вертикальных дивизоров Картье. Существует точная последовательность пучков (в этальной топологии схемы X)

$$0 \to \mathbb{G}_{m,X} \to h_*\mathbb{G}_{m,V} \to \mathrm{Div}_X^{\mathrm{vert}} \to 0, \tag{1}$$

где $h : V \hookrightarrow X$ – вложение общего схемного слоя морфизма π [3, последняя формула на с. 637].

Мы имеем

$$\mathrm{Div}_X = \bigoplus_{\substack{y\in X\\ \mathrm{codim}_X(y)=1}} i_{y*}\mathbb{Z} = \left(\bigoplus_{\substack{y\in V\\ \mathrm{codim}_X(y)=1}} i_{y*}\mathbb{Z}\right) \bigoplus \mathrm{Div}_X^{\mathrm{vert}}.$$

Хорошо известно, что $H^1(X, \mathrm{Div}_X) = 0$ [4, гл. 3, § 2, пример 2.22]. Следовательно, $H^1(X, \mathrm{Div}_X^{\mathrm{vert}}) = 0$. Значит, (1) даёт точную последовательность

$$\begin{aligned} 0 \to H^2(X, \mathbb{G}_m) \to H^2(X, h_*\mathbb{G}_{m,V}) \to H^2(X, \mathrm{Div}_X^{\mathrm{vert}}) \\ \to H^3(X, \mathbb{G}_m) \to H^3(X, h_*\mathbb{G}_{m,V}) \to H^3(X, \mathrm{Div}_X^{\mathrm{vert}}). \end{aligned} \tag{2}$$

Спектральная последовательность Лере

$$E_2^{p,q} = H^p(X, R^q h_*\mathbb{G}_{m,V}) \Rightarrow H^{p+q}(V, \mathbb{G}_{m,V})$$

даёт точную последовательность

$$0 \to E_2^{1,0} \to E^1 \to E_2^{0,1} \xrightarrow{d_2^{0,1}} E_2^{2,0} \to E_1^2 \to E_2^{1,1} \to E_2^{3,0},$$

где $E_1^2 = \mathrm{Ker}[E^2 \to E_2^{0,2}]$ [4, приложение B]. Следовательно, мы имеем точную последовательность

$$\begin{aligned} 0 \to H^1(X, h_*\mathbb{G}_{m,V}) \to H^1(V, \mathbb{G}_{m,V}) \\ \to H^0(X, R^1h_*\mathbb{G}_{m,V}) \xrightarrow{d_2^{0,1}} H^2(X, h_*\mathbb{G}_{m,V}) \\ \to \mathrm{Ker}[H^2(V, \mathbb{G}_{m,V}) \to H^0(X, R^2h_*\mathbb{G}_{m,V})] \\ \to H^1(X, R^1h_*\mathbb{G}_{m,V}). \end{aligned} \tag{3}$$

С другой стороны, $R^1h_*\mathbb{G}_{m,V} = 0$ в силу аргументов п. (b) доказательства леммы 4.4.1 в [3]. Поэтому (3) даёт изоморфизм

$$H^2(X, h_*\mathbb{G}_{m,V}) \xrightarrow{\sim} \mathrm{Ker}[\mathrm{Br}'(V) \to H^0(X, R^2h_*\mathbb{G}_{m,V})]$$

и (2) даёт точную последовательность

$$0 \to \mathrm{Br}'(X) \to \mathrm{Ker}[\mathrm{Br}'(V) \to H^0(X, R^2h_*\mathbb{G}_{m,V})] \to H^2(X, \mathrm{Div}_X^{\mathrm{vert}}). \tag{4}$$

В дальнейшем мы обозначаем через η общую схемную точку кривой C и через $\kappa(y)^{\mathrm{s}}$ сепарабельное замыкание поля вычетов $\kappa(y)$ в алгебраическом замыкании $\overline{\kappa(y)}$.

Существует каноническое отображение $\mathrm{Br}(k) \to \mathrm{Br}'(V)$, индуцированное структурным морфизмом $V \to \operatorname{Spec} k$. С другой стороны, существует каноническая точная последовательность

$$0 \to \mathrm{Br}(C) \to \mathrm{Br}(k) \to \bigoplus_{\substack{v \in C \\ v \neq \eta}} \mathrm{Hom}_{\mathrm{cont}}(\mathrm{Gal}(\kappa(v)^{\mathrm{s}}/\kappa(v)), \mathbb{Q}/\mathbb{Z}) \\ \to H^3(C, \mathbb{G}_m) \to H^3(\operatorname{Spec}(k), \mathbb{G}_m)$$

[4, гл. III, § 2, пример 2.22(a)], где $\mathrm{Hom}_{\mathrm{cont}}$ обозначает группу непрерывных гомоморфизмов. В нашем случае C – полная гладкая алгебраическая кривая над конечным полем F_q, поэтому $\mathrm{Br}(C) = 0$ и $H^3(C, \mathbb{G}_m) = \mathbb{Q}/\mathbb{Z}$ [4, гл. III, § 2, пример 2.22(g)]. Мы приходим к хорошо известной точной последовательности глобальной теории полей классов

$$0 \to \mathrm{Br}(k) \to \bigoplus_{\substack{v \in C \\ v \neq \eta}} \mathrm{Hom}_{\mathrm{cont}}(\mathrm{Gal}(\kappa(v)^{\mathrm{s}}/\kappa(v)), \mathbb{Q}/\mathbb{Z}) \to \mathbb{Q}/\mathbb{Z}. \tag{5}$$

Для замкнутой точки $v \in C$ рассмотрим каноническое вложение $i_v : \operatorname{Spec} \kappa(v) \hookrightarrow C$. Спектральная последовательность Лере

$$E_2^{p,q} = H^p(C, R^q i_{v*}\mathbb{Z}) \to H^{p+q}(\operatorname{Spec} \kappa(v), \mathbb{Z})$$

дает точную последовательность

$$0 \to E_2^{1,0} \to E^1 \to E_2^{0,1} \xrightarrow{d_2^{0,1}} E_2^{2,0} \to E_1^2 \to E_2^{1,1},$$

где $E_1^2 = \mathrm{Ker}[E^2 \to E_2^{0,2}]$ [4, приложение B]; следовательно, мы имеем точную последовательность

$$\begin{aligned} 0 &\to H^1(C, i_{v*}\mathbb{Z}) \to H^1(\operatorname{Spec} \kappa(v), \mathbb{Z}) \\ &\to H^0(C, R^1 i_{v*}\mathbb{Z}) \xrightarrow{d_2^{0,1}} H^2(C, i_{v*}\mathbb{Z}) \\ &\to \mathrm{Ker}[H^2(\operatorname{Spec} \kappa(v), \mathbb{Z}) \to H^0(C, R^2 i_{v*}\mathbb{Z})] \\ &\to H^1(C, R^1 i_{v*}\mathbb{Z}). \end{aligned} \tag{6}$$

Хорошо известно, что $R^q i_{v*}\mathbb{Z} = 0$ для всех $q > 0$ [4, гл. III, § 2, пример 2.22(a)]; поэтому (6) даёт изоморфизм

$$H^2(C, i_{v*}\mathbb{Z}) \xrightarrow{\sim} H^2(\operatorname{Spec} \kappa(v), \mathbb{Z}).$$

С другой стороны,

$$H^2(\operatorname{Spec} \kappa(v), \mathbb{Z}) \xrightarrow{\sim} \mathrm{Hom}_{\mathrm{cont}}(\mathrm{Gal}(\kappa(v)^{\mathrm{s}}/\kappa(v)), \mathbb{Q}/\mathbb{Z})$$

[4, гл. III, § 2, пример 2.22]. Следовательно,

$$H^2(C,\mathrm{Div}_C)\overset{\mathrm{def}}{=}H^2\left(C,\bigoplus_{\substack{v\in C\\ v\neq\eta}}i_{v*}\mathbb{Z}\right)$$
$$=\bigoplus_{\substack{v\in C\\ v\neq\eta}}H^2(C,i_{v*}\mathbb{Z})$$
$$=\bigoplus_{\substack{v\in C\\ v\neq\eta}}\mathrm{Hom}_{\mathrm{cont}}(\mathrm{Gal}(\kappa(v)^{\mathrm{s}}/\kappa(v)),\mathbb{Q}/\mathbb{Z})$$

[4, гл. III, § 2, пример 2.22(a)]. Поэтому (5) даёт точную последовательность

$$0\to\mathrm{Br}(k)\to H^2(C,\mathrm{Div}_C). \tag{7}$$

С другой стороны, имеется канонический морфизм

$$\pi^*:H^*(C,\mathrm{Div}_C)\to H^*(X,\pi^*(\mathrm{Div}_C)).$$

Каноническое вложение $\pi^*(\mathrm{Div}_C)\hookrightarrow\mathrm{Div}_X^{\mathrm{vert}}$ даёт каноническое отображение

$$H^2(X,\pi^*(\mathrm{Div}_C))\to H^2(X,\mathrm{Div}_X^{\mathrm{vert}}).$$

Следовательно, имеются канонические морфизмы

$$\varphi:\mathrm{Br}(k)\to\mathrm{Br}'(V), \tag{8}$$

$$H^2(C,\mathrm{Div}_C)\to H^2(X,\mathrm{Div}_X^{\mathrm{vert}}). \tag{9}$$

Пусть $B=\mathrm{Ker}[\mathrm{Br}'(V)\to H^0(X,R^2h_*\mathbb{G}_{m,V})]$. Очевидно, что (4), (7) – (9) дают коммутативную диаграмму с точными строками

$$\begin{array}{ccccccc}0&\longrightarrow&\mathrm{Br}'(X)&\longrightarrow&B&\longrightarrow&H^2(X,\mathrm{Div}_X^{\mathrm{vert}})\\ &&\uparrow&&\varphi\uparrow&&\uparrow\\ &&0&\longrightarrow&\mathrm{Br}(k)\cap\varphi^{-1}(B)&\longrightarrow&H^2(C,\mathrm{Div}_C)\end{array} \tag{10}$$

По условию теоремы 1 для простого числа l, не делящего $\mathrm{Card}([\mathrm{NS}(V)]_{\mathrm{tors}})$ и отличного от характеристики поля $\mathbb{F}_q$, верна гипотеза Тэйта для дивизоров на V

$$\mathrm{NS}(V)\otimes\mathbb{Q}_l\xrightarrow{\sim}[H^2(V\otimes k^{\mathrm{s}},\mathbb{Q}_l(1))]^{\mathrm{Gal}(k^{\mathrm{s}}/k)}.$$

Поэтому в силу [5, теорема 2.2] группа $\mathrm{Br}'(V\otimes_k k^{\mathrm{s}})^{\mathrm{Gal}(k^{\mathrm{s}}/k)}(l)$ конечна.

Хорошо известно, что спектральная последовательность Хохшильда–Серра

$$H^p(\mathrm{Gal}(k^{\mathrm{s}}/k),H^q_{\text{ét}}(V\otimes_k k^{\mathrm{s}},\mathbb{G}_m))\Longrightarrow H^{p+q}_{\text{ét}}(V,\mathbb{G}_m)$$

даёт точную последовательность

$$0\to E_2^{1,0}\to E^1\to E_2^{0,1}\xrightarrow{d_2^{0,1}}E_2^{2,0}\to E_1^2\to E_2^{1,1}\to E_2^{3,0},$$

где $E_1^2 = \mathrm{Ker}[E^2 \to E_2^{0,2}]$; очевидно, что эта последовательность имеет вид

$$\begin{aligned} 0 \to \mathrm{Pic}(V) &\to \mathrm{Pic}(V \otimes_k k^{\mathrm{s}})^{\mathrm{Gal}(k^{\mathrm{s}}/k)} \\ &\to \mathrm{Br}(k) \to \mathrm{Ker}[\mathrm{Br}'(V) \to \mathrm{Br}'(V \otimes_k k^{\mathrm{s}})^{\mathrm{Gal}(k^{\mathrm{s}}/k)}] \\ &\to H^1(\mathrm{Gal}(k^{\mathrm{s}}/k), \mathrm{Pic}(V \otimes_k k^{\mathrm{s}})) \to H^3(\mathrm{Gal}(k^{\mathrm{s}}/k), \mathbb{G}_m). \end{aligned}$$

Поэтому из конечности группы $\mathrm{Br}'(V \otimes_k k^{\mathrm{s}})^{\mathrm{Gal}(k^{\mathrm{s}}/k)}(l)$ и тривиальности группы $H^1(\mathrm{Gal}(k^{\mathrm{s}}/k), \mathrm{Pic}(V \otimes_k k^{\mathrm{s}}))(l)$ (так как l не делит порядок группы кручения в $\mathrm{Pic}(V \otimes_k k^{\mathrm{s}})$) следует, что группа $[\mathrm{Br}'(V)/\operatorname{Im}(\mathrm{Br}(k) \to \mathrm{Br}'(V))](l)$ конечна.

Значит, группа

$$B/[\varphi(\mathrm{Br}(k)) \cap B](l) = \mathrm{Coker}[\mathrm{Br}(k) \cap \varphi^{-1}(B) \xrightarrow{\varphi} B](l)$$

конечная. Поэтому достаточно показать, что пересечение группы $\mathrm{Br}'(X)(l)$ с образом морфизма групп

$$[\mathrm{Br}(k) \cap \varphi^{-1}(B)](l) \xrightarrow{\varphi} B(l)$$

конечно. В силу коммутативности диаграммы (10) достаточно доказать конечность ядра канонического отображения $H^2(C, \mathrm{Div}_C) \to H^2(X, \mathrm{Div}_X^{\mathrm{vert}})$.

Очевидно, что ядро отображения $H^2(C, \mathrm{Div}_C) \to H^2(X, \mathrm{Div}_X^{\mathrm{vert}})$ является прямой суммой по всем замкнутым точкам v кривой C ядер отображений

$$\pi^* : H^2(\operatorname{Spec} \kappa(v), \mathbb{Z}) \to \bigoplus_D H^2(\operatorname{Spec} \kappa(D), \mathbb{Z}), \tag{11}$$

которые также можно записать в виде

$$\mathrm{Hom}_{\mathrm{cont}}(\mathrm{Gal}(\overline{\kappa(v)}/\kappa(v)), \mathbb{Q}/\mathbb{Z}) \to \bigoplus_D \mathrm{Hom}_{\mathrm{cont}}(\mathrm{Gal}(\kappa(D)^{\mathrm{s}}/\kappa(D)), \mathbb{Q}/\mathbb{Z}).$$

Здесь D пробегает неприводимые компоненты слоя морфизма $X \to C$ над точкой v, поле $\kappa(D)^{\mathrm{s}}$ является сепарабельным замыканием поля $\kappa(D)$ рациональных функций на D.

Пусть κ_D – целое замыкание $\kappa(v)$ в поле $\kappa(D)$. Тогда $\kappa(v) \subset \kappa(D)$ – такое расширение полей, что многообразие D является геометрически целым над κ_D. Мы имеем композицию канонических морфизмов полей

$$\kappa(D) = \kappa_D(D) \leftarrow \kappa_D \leftarrow \kappa(v),$$

индуцирующую композицию отображений

$$H^2(\operatorname{Spec} \kappa(v), \mathbb{Z}) \to H^2(\operatorname{Spec} \kappa_D, \mathbb{Z}) \to H^2(\operatorname{Spec} \kappa_D(D), \mathbb{Z}) = H^2(\operatorname{Spec} \kappa(D), \mathbb{Z}).$$

Очевидно, что композиция

$$H^2(\operatorname{Spec} \kappa(v), \mathbb{Z}) \xrightarrow{\pi^*} \bigoplus_D H^2(\operatorname{Spec} \kappa(D), \mathbb{Z}) \to H^2(\operatorname{Spec} \kappa(D), \mathbb{Z}) \tag{12}$$

отображения π^* в (11) и канонической проекции

$$\bigoplus_D H^2(\operatorname{Spec} \kappa(D), \mathbb{Z}) \to H^2(\operatorname{Spec} \kappa(D), \mathbb{Z})$$

является композицией канонических отображений

$$H^2(\operatorname{Spec}\kappa(v),\mathbb{Z})\to H^2(\operatorname{Spec}\kappa_D,\mathbb{Z})\to H^2(\operatorname{Spec}\kappa_D(D),\mathbb{Z})=H^2(\operatorname{Spec}\kappa(D),\mathbb{Z})$$

$$\xrightarrow{x\mapsto \operatorname{mult}_{\pi^{-1}(v)}(D)\cdot x} H^2(\operatorname{Spec}\kappa(D),\mathbb{Z}), \tag{13}$$

где $\operatorname{mult}_{\pi^{-1}(v)}(D)$ – кратность подмногообразия D в слое $X_v=\pi^{-1}(v)$.

Пусть $\mathbb{F}_q$ – конечное поле порядка q и пусть $W\hookrightarrow\mathbb{P}^n_{\mathbb{F}_q}$ – геометрически неприводимое проективное подмногообразие размерности r и степени d. Хорошо известный результат Ленга и Вейля [6, теорема 1] показывает, что

$$|\operatorname{Card}(W(\mathbb{F}_q))-q^r|\le(d-1)(d-2)\,q^{r-1/2}+c(n,d,r)\,q^{r-1}$$

для константы $c(n,d,r)>0$, зависящей только от n,d и r.

В силу оценки Ленга–Вейля и леммы Гензеля многообразие V имеет точки почти во всех пополнениях глобального поля k. Другими словами, имеется такое конечное множество S замкнутых точек кривой C, что для всех $v\notin S$ морфизм $X\to C$, ограниченный на спектр локального кольца $\mathcal{O}_v$, имеет сечение $\theta:\operatorname{Spec}\mathcal{O}_v\to X\times_C\operatorname{Spec}\mathcal{O}_v$. В силу следствия 2.2 в [7], точка $\theta(v)$ является регулярной точкой схемного слоя $\pi^{-1}(v)$. Поэтому $\mathcal{O}_{\pi^{-1}(v),\theta(v)}$ – регулярное локальное кольцо и слой $\pi^{-1}(v)$ аналитически неприводим в точке $\theta(v)$ [8, гл. 11, замечание 1 к предложению 11.24]; в частности, $\theta(\operatorname{Spec}\mathcal{O}_v)$ пересекает *одну* неприводимую компоненту схемного слоя $\pi^{-1}(v)$ (то, что сечение на регулярной модели пересекает в точности одну неприводимую компоненту кратности 1 слоя, доказано также в [9, лемма 1.1, b]). Поэтому любой вертикальный дивизор Картье D с носителем в слое $\pi^{-1}(v)$ может быть единственным образом записан в виде

$$D=n_0\cdot\pi^{-1}(v)+\sum_i n_i\cdot D_i,$$

где $n_j\in\mathbb{Z}$ и D_i – такие неприводимые компоненты слоя $\pi^{-1}(v)$, что выполнено равенство $D_i\cap\theta(\operatorname{Spec}\mathcal{O}_v)=\varnothing$. Следовательно, получаем разложение пучков

$$\operatorname{Div}^{\operatorname{vert}}_{X\times_C\operatorname{Spec}\mathcal{O}_v}=\pi^*(\operatorname{Div}_{\operatorname{Spec}\mathcal{O}_v})\oplus\left(\bigoplus_{\substack{y\in X\times_C\operatorname{Spec}\mathcal{O}_v\setminus V,\ \operatorname{codim}_{X\times_C\operatorname{Spec}\mathcal{O}_v}(y)=1,\\ \overline{\{y\}}\cap\theta(\operatorname{Spec}\mathcal{O}_v)=\varnothing}} i_{y*}\mathbb{Z}\right). \tag{14}$$

Соотношение $\pi\circ\theta=\mathrm{id}_{\operatorname{Spec}\mathcal{O}_v}$ показывает, что композиция

$$H^*(\operatorname{Spec}\mathcal{O}_v,\operatorname{Div}_{\operatorname{Spec}\mathcal{O}_v})\xrightarrow{\pi^*}H^*(X\times_C\operatorname{Spec}\mathcal{O}_v,\pi^*(\operatorname{Div}_{\operatorname{Spec}\mathcal{O}_v}))$$
$$\xrightarrow{\theta^*}H^*(\operatorname{Spec}\mathcal{O}_v,\operatorname{Div}_{\operatorname{Spec}\mathcal{O}_v})$$

является тождественным отображением. Поэтому мы получаем вложение

$$H^2(\operatorname{Spec}\mathcal{O}_v,\operatorname{Div}_{\operatorname{Spec}\mathcal{O}_v})\overset{\pi^*}{\hookrightarrow}H^2(X\times_C\operatorname{Spec}\mathcal{O}_v,\pi^*(\operatorname{Div}_{\operatorname{Spec}\mathcal{O}_v})).$$

С другой стороны, разложение (14) даёт вложение

$$H^2(X \times_C \operatorname{Spec} \mathcal{O}_v, \pi^*(\operatorname{Div}_{\operatorname{Spec} \mathcal{O}_v})) \hookrightarrow H^2(X \times_C \operatorname{Spec} \mathcal{O}_v, \operatorname{Div}^{\operatorname{vert}}_{X \times_C \operatorname{Spec} \mathcal{O}_v})).$$

Следовательно, для $v \notin S$ ядро отображения (11) тривиально.

Остаётся доказать, что для $v \in S$ ядро композиции (12) конечно.

Для начала заметим, что отображение $H^2(\operatorname{Spec} \kappa_D, \mathbb{Z}) \to H^2(\operatorname{Spec} \kappa(D), \mathbb{Z})$ в композиции (13) инъективно. Действительно, это отображение совпадает с отображением

$$H^1(\operatorname{Gal}(\overline{\kappa_D}/\kappa_D), \mathbb{Q}/\mathbb{Z}) \to H^1(\operatorname{Gal}(\kappa(D)^{\mathrm{s}}/\kappa(D)), \mathbb{Q}/\mathbb{Z}).$$

Оно инъективно, потому что $\kappa(D)$ и любое алгебраическое замыкание поля κ_D линейно разделены, так что имеется канонический изоморфизм [10, гл. V, § 10, п. 4, теорема 1]

$$\operatorname{Gal}(\overline{\kappa_D}/\kappa_D) \xrightarrow{\sim} \operatorname{Gal}(\overline{\kappa_D}(D)/\kappa(D));$$

остаётся использовать каноническую последовательность инфляции – ограничения [11, гл. IV, § 5, предложение 5.1]

$$\begin{aligned} 0 \to H^1(\operatorname{Gal}(\overline{\kappa_D}(D)/\kappa(D)), \mathbb{Q}/\mathbb{Z}) &\xrightarrow{\inf} H^1(\operatorname{Gal}(\kappa(D)^{\mathrm{s}}/\kappa(D)), \mathbb{Q}/\mathbb{Z}) \\ &\xrightarrow{\operatorname{res}} H^1(\operatorname{Gal}(\kappa(D)^{\mathrm{s}}/\overline{\kappa_D}(D)), \mathbb{Q}/\mathbb{Z}). \end{aligned}$$

Следовательно, достаточно доказать конечность ядра композиции отображений

$$H^2(\operatorname{Spec} \kappa(v), \mathbb{Z}) \to H^2(\operatorname{Spec} \kappa_D, \mathbb{Z}) \xrightarrow{x \mapsto \operatorname{mult}_{\pi^{-1}(v)}(D) \cdot x} H^2(\operatorname{Spec} \kappa_D, \mathbb{Z}). \qquad (15)$$

Пусть $d = [\kappa_D : \kappa(v)]$. Поскольку композиция ограничения и кограничения является умножением на d, то мы видим, что ядро композиции (15) содержится в ядре умножения на $\operatorname{mult}_{\pi^{-1}(v)}(D) \cdot d : \mathbb{Q}/\mathbb{Z} \to \mathbb{Q}/\mathbb{Z}$, которое, очевидно, конечно. Теорема 1 доказана.

Теорема 2. *Пусть $\pi : X \to C$ – сюръективный морфизм гладких проективных многообразий над конечным полем $\mathbb{F}_q$ нечетной характеристики p, общий схемный слой которого является* K3*–поверхностью V над полем $k = \kappa(C)$ рациональных функций кривой C. Предположим, что $\operatorname{NS}(V) = \operatorname{NS}(V \otimes \overline{k})$. Тогда для любого простого числа $l \neq p$*

$$\operatorname{NS}(X) \otimes \mathbb{Q}_l \xrightarrow{\sim} [H^2(X \otimes \overline{\mathbb{F}}_q, \mathbb{Q}_l(1))]^{\operatorname{Gal}(\overline{\mathbb{F}}_q/\mathbb{F}_q)}.$$

Действительно, для K3 – поверхности V верна гипотеза Тэйта для дивизоров [12], поэтому теорема следует из предложений $1 - 2$ и теоремы 1.

Список литературы / References

[1] J.S. Milne, “Values of zeta functions of varieties over finite fields”, *Amer. J. Math.*, **108** (1986), 297–360.

[2] J. Tate, “Conjectures on algebraic cycles in l-adic cohomology”, *Proc. Symposia in Pure Math.*, **55** (1994 Part 1), 71 – 83.

[3] Colliot-Thélène J.-L., Skorobogatov A.N., Swinnerton-Dyer P., “Hasse principle for pencils of curves of genus one whose Jacobians have rational 2-division points”, *Invent. Math.*, **134**:3 (1998), 579–650.

[4] Милн Дж., *Этальные когомологии*, Мир, М., 1983; [Milne J.S., *Etale cohomology*, Princeton Univ. Press, Princeton, 1980].

[5] Танкеев С. Г., “О группе Брауэра арифметической модели гиперкэлерова многообразия над числовым полем”, *Изв. РАН. Сер. матем.*, **79**:3 (2015), 203 – 224; [Tankeev S.G., “On the Brauer group of arithmetic model of a hyperkähler variety over a number field”, *Izv. Math.*, **79**:3 (2015), 623–644].

[6] Lang S., Weil A., “Number of points of varieties in finite fields”, *Amer. J. Math.*, **76**:4 (1954), 819–827.

[7] Танкеев С. Г., “О группе Брауэра арифметической схемы. II”, *Изв. РАН. Сер. матем.*, **67**:5 (2003), 155–176; [Tankeev S.G., “On the Brauer group of arithmetic scheme. II”, *Izv. Math.*, **67**:5 (2003), 1007–1029].

[8] Атья М., Макдональд И., *Введение в коммутативную алгебру*, Мир, М., 1972; [Atiyah M.F., Macdonald I.G., *Introduction to commutative algebra*, Addison–Wesley Publ. Co., Massachusets, 1969].

[9] Skorobogatov A. N., “Descent on fibrations over the projective line”, *Amer. J. Math.*, **118**:5 (1996), 905–923.

[10] Бурбаки Н., *Алгебра. Многочлены и поля. Упорядоченные группы, Элементы математики*, Наука, М., 1965; [Bourbaki N., *Éléments de Mathématique. Algébre, livre II*, Hermann, Paris, 1963].

[11] *Алгебраическая теория чисел*, ред. Касселс Дж., Фрёлих А., Мир, М., 1969; [*Algebraic number theory*, Proc. Internat. Conf. Brighton, 1965, eds. Cassels G. W. S., Frölich A., Academic Press, London, and Thompson, Washington, DC, 1967].

[12] Madapusi Pera K., “The Tate conjecture for K 3 surfaces in odd characteristic Descent on fibrations over the projective line”, *Invent. math.*, **201** (2015), 625–668.

Prokhorova T. V., "On the Tate Conjectures for Divisors on a Fibred Variety and on its Generic Scheme Fibre in the Case of Finite Characteristic", *Modeling and Analysis of Information Systems*, **24**:2 (2017), 205–214.

DOI: 10.18255/1818-1015-2017-2-205-214

Abstract. We investigate interrelations between the Tate conjecture for divisors on a fibred variety over a finite field and the Tate conjecture for divisors on the generic scheme fibre under the condition that the generic scheme fibre has zero irregularity. Let $\pi : X \to C$ be a surjective morphism of smooth projective varieties over a finite field $\mathbb{F}_q$ of characteristic p, C is a curve and the generic scheme fibre of π is a smooth variety V over the field $k = \kappa(C)$ of rational functions of the curve C, $\overline{k}$ is an algebraic closure of the field k, k^s is its separable closure, $\mathrm{NS}(V)$ is the Néron - Severi group of classes of divisors on the variety V modulo algebraic equivalence, and assume that the following conditions hold: $H^1(V \otimes \overline{k}, \mathcal{O}_{V\otimes\overline{k}}) = 0$, $\mathrm{NS}(V) = \mathrm{NS}(V \otimes \overline{k})$. If, for a prime number l not dividing $\mathrm{Card}([\mathrm{NS}(V)]_{\mathrm{tors}})$ and different from the characteristic of the field $\mathbb{F}_q$, the following relation holds $\mathrm{NS}(V) \otimes \mathbb{Q}_l \overset{\sim}{\to} [H^2(V \otimes k^{\mathrm{s}}, \mathbb{Q}_l(1))]^{\mathrm{Gal}(k^{\mathrm{s}}/k)}$ (in other words, if the Tate conjecture for divisors on V

holds), then for any prime number $l \neq \mathrm{char}(\mathbb{F}_q)$ the Tate conjecture holds for divisors on X: $\mathrm{NS}(X) \otimes \mathbb{Q}_l \xrightarrow{\sim} [H^2(X \otimes \overline{\mathbb{F}}_q, \mathbb{Q}_l(1))]^{\mathrm{Gal}(\overline{\mathbb{F}}_q/\mathbb{F}_q)}$. In particular, it follows from this result that the Tate conjecture for divisors on an arithmetic model of a K 3 surface over a sufficiently large global field of finite characteristic different from 2 holds as well.

Keywords: Tate conjecture, global field, Brauer group, arithmetic model, K 3 surface

About the authors:
Tatyana V. Prokhorova, orcid.org/0000-0002-6883-2087, PhD,
A. G. and N. G. Stoletov Vladimir State University,
87 Gorky str., Vladimir 600000, Russia,
e-mail: tvprokhorova@mail.ru

Моделирование и анализ информационных систем. Т. 24, № 2 (2017), с. 215–226
Modeling and Analysis of Information Systems. Vol. 24, No 2 (2017), pp. 215–226

DOI: 10.18255/1818-1015-2017-2-215-226

УДК 004.056.3

Дедубликация в системе резервного копирования с хранением информации в базе данных

Таранин С. М.

получена 18 сентября 2016

Аннотация. Профилактика потери данных с цифровых носителей включает такой процесс, как резервное копирование. Он может проводиться вручную простым копированием данных на внешние носители или автоматизированно по расписанию с помощью специальных программных средств. Существуют системы удаленного резервного копирования, когда данные сохраняются по сети в удаленное хранилище. Такие системы являются многопользовательскими и обрабатывают большие объемы данных. В общем хранилище могут встретиться файлы, содержащие одинаковые фрагменты. Для исключения повторяющихся данных применяется механизм дедубликации (англ. de-duplication). Он представляет собой метод сжатия информации, когда поиск копий производится по всему массиву данных, а не в пределах одного файла. Главным преимуществом использования данной технологии является существенная экономия дискового пространства. Однако механизм исключения повторяющихся данных может существенно снизить скорость сохранения и восстановления информации. Настоящая статья посвящена проблеме реализации такого механизма в системе резервного копирования с хранением информации в реляционной базе данных. В данной работе рассматривается пример реализации такой системы, работающей в двух режимах: с дедубликацией данных и без нее. В статье приведен пример схемы классов для разработки клиентской части приложения, а также описание таблиц и связей между ними в базе данных, что относится к серверной части. Далее автор предлагает алгоритм сохранения данных с дедубликацией, а также приводит результаты сравнительных тестов скорости работы алгоритмов сохранения и восстановления информации при работе с реляционными системами управления базами данных разных производителей.

Ключевые слова: файл, данные, резервное копирование, дедубликация, база данных

Для цитирования: Таранин С. М., "Дедубликация в системе резервного копирования с хранением информации в базе данных", *Моделирование и анализ информационных систем*, **24**:2 (2017), 215–226.

Об авторах:
Таранин Сергей Максимович, orcid.org/0000-0001-8117-7358, аспирант,
Ярославский государственный университет им. П.Г. Демидова,
ул. Советская, 14, г. Ярославль, 150003 Россия, e-mail: staranin0208@yandex.ru

Введение

Одной из угроз целостности данных на автоматизированном рабочем месте (АРМ) является выход из строя цифрового энергонезависимого устройства хранения информации. Примерами таких устройств являются жесткие диски (англ. hard disk

drive HDD), твердотельные накопители (англ. solid-state drive, SSD), а также гибридные устройства (англ. solid-state hybrid drive, SSHD), представляющие собой компромиссное решение между стоимостью первых и производительностью последних. Нарушение целостности данных на цифровых носителях может произойти вследствие различных причин. Главным фактором, ограничивающим длительность хранения информации на цифровых носителях, является срок их службы. Он может быть разным в зависимости от производителя, однако даже у самых качественных и дорогих устройств срок службы составляет в среднем пять лет.

Профилактика потери данных с цифровых носителей включает такой процесс, как резервное копирование [2]. Он может проводиться вручную простым копированием данных на внешние носители или автоматически по расписанию с помощью специальных программных средств. Существуют системы удаленного резервного копирования, когда данные сохраняются по сети в удаленное хранилище. Такие системы являются многопользовательскими и обрабатывают большие объемы данных. В общем хранилище могут встретиться файлы, содержащие одинаковые фрагменты. Для исключения повторяющихся данных применяется механизм дедубликации (англ. de-duplication)[7,8]. Он представляет собой метод сжатия информации, когда поиск копий производится по всему массиву данных, а не в пределах одного файла. Главным преимуществом использования данной технологии является существенная экономия дискового пространства. Однако при этом снижается производительность системы резервного копирования.

Настоящая статья посвящена проблеме реализации механизма исключения дублирования данных в системе резервного копирования с хранением информации в базе данных (БД). В рамках данной работы рассмотрено использование реляционных систем управления базами данных (СУБД), что связано с их широкой распространенностью и встроенными механизмами обеспечения целостности [10,11,12]. В ходе работы будет предложена схема реализации клиентской и серверной части системы [9], алгоритм сохранения данных с дедубликацией. Также будут приведены и проанализированы результаты сравнительных тестов производительности системы с включенным и выключенным механизмом исключения дублирования данных при работе с СУБД разных производителей.

1. Реализация системы резервного копирования

В работе [1] описан подход к реализации системы резервного копирования с хранением в БД без дедубликации. В таком случае для хранения пользовательских данных на сервере предлагается завести в БД две таблицы $Model$ и $Data$. В первой хранится информация об объекте файловой системы, а во второй – данные этого объекта, разбитые на блоки фиксированной длины. Связь между таблицами $Model$ и $Data$ «один ко многим» соответственно. Таким образом, на каждый блок данных указывает только один файл.

Для исключения повторяющихся данных в таблице $Data$, ее связь с таблицей $Model$ должна быть «многие ко многим», когда на один блок данных могут ссылаться несколько файлов. Для организации такой связи предлагается ввести дополнительную таблицу $Link$.

Описание таблицы *Model* остается как в первоначальном варианте [1], а из таблицы *Data* удалено поле $FILEID$. Таблица *Link* содержит следующие поля:

1. ID Первичный ключ.
2. FILEID Внешний ключ. Идентификатор записи в таблице *Model*.
3. DATAID Внешний ключ. Идентификатор записи в таблице *Data*.
4. ORD Порядковый номер блока с идентификатором $DATAID$ в файле с идентификатором $FILEID$.

Порядок блоков внутри файла обеспечивается за счет значений ORD. Следующий SQL-запрос по идентификатору записи таблицы *Model* вернет последовательность блоков в том порядке, в котором они идут внутри файла:

```
select Data.BLOCK
   from Link
   inner join Data
   on Link.FILEID = [идентификатор] and Link.DATAID = Data.ID
order by Link.ORD;
```

Клиентская часть системы резервного копирования представляет собой приложение, которое должно уметь читать и записывать содержимое пользовательских файлов, взаимодействовать с СУБД путем выполнения *SQL*-запросов, а также шифровать данные при необходимости. Выделим основные компоненты клиентского приложения:

1. Ядро. Содержит реализацию классов для представления пользовательских данных в виде набора записей таблицы, а также описание интерфейсов для взаимодействия с СУБД и файловой системой пользователя.
2. Контроллер для взаимодействия с СУБД. Реализует интерфейс для взаимодействия с СУБД. Таких контроллеров может быть несколько, по количеству поддерживаемых СУБД разных производителей. Общую часть, которая одинакова для всех СУБД, можно вынести в базовый класс.
3. Контроллер для работы с файловой системой [15]. Реализует интерфейс для взаимодействия с файловой системой пользователя, реализацию алгоритмов сохранения (с дедубликацией и без нее), восстановления данных, а также сканирования директории на наличие изменений.
4. Криптопровайдер. Реализует алгоритмы шифрования и хеширования данных.
5. Модель копируемой директории. Является представлением набора записей таблицы *Model* на стороне клиента.

На рисунке 1 представлен пример схемы классов для реализации клиентского приложения.

Ядро системы составляют классы *Record* и *RecordSet*. Класс *Record* необходим для представления данных на стороне клиента в виде записи, а *RecordSet* – в виде упорядоченного набора записей.

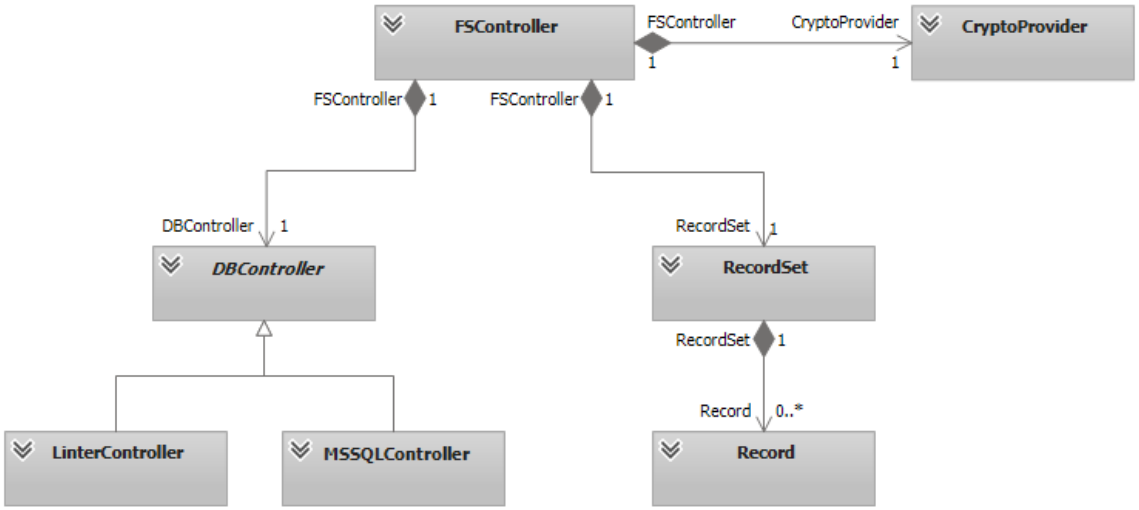


Рис. 1: Схема классов клиентского приложения

Fig. 1. The classes schem of client application

Классы *MSSQLController* и *LinterController* представляют собой реализацию контроллеров для взаимодействия с СУБД MSSQL и Linter соответственно. При необходимости можно расширять список поддерживаемых СУБД, реализуя подобные классы по некоторому общему интерфейсу. Такой интерфейс может включать функции сохранения, обновления, удаления и чтения записей таблиц по идентификатору, функции удаления и создания таблиц, функцию выборки записей по значению полей и другие. Абстрактный класс *DBController* содержит описание общего интерфейса взаимодействия с СУБД и функции, реализация которых одинакова для всех поддерживаемых систем. Например, функция удаления таблицы по имени для всех СУБД выполняет *SQL*-запрос вида:

```
drop table [имя_таблицы];
```

Класс *CryptoProvider* содержит реализацию алгоритмов шифрования хеширования данных.

Класс *FSController* представляет собой реализацию контроллера для работы с файловой системой пользователя. Он содержит актуальную копию таблицы *Model* в виде объекта типа *RecordSet*, экземпляр класса контроллера для взаимодействия с СУБД и экземпляр класса криптопровайдера.

1.1. Сохранение и восстановление данных

Предлагается следующий рекурсивный алгоритм сохранения данных с дедубликацией:

1 Для каждого файла из текущей директории:

1.1 Создаем запись для таблицы *Model* и сохраняем ее *insert*-запросом к БД.

1.2 Получаем идентификатор сохраненной записи.

1.3 Сохраняем запись в таблице *Model* на стороне клиента.

1.4 Читаем файл с диска блоками, формируя массив записей для сохранения в таблицу *Data*.

1.5 Для каждого блока сохраняемого файла:

1.5.1 Считаем хеш-код блока

1.5.2 Если в таблице *Data* нет блока с таким хеш-кодом, сохраняем новую запись в таблицу *Data* *insert*-запросом к БД.

1.5.3 Создаем запись для таблицы *Link* и сохраняем ее *insert*-запросом к БД.

2 Для каждой папки из текущей директории:

2.1 Формируем запись для таблицы *Model* и сохраняем ее *insert*-запросом к БД.

2.2 Запускаем данный алгоритм для файлов в этой директории.

3 Считаем хэш-код для текущей директории и сохраняем его в поле *HASH* соответствующей записи таблицы *Model* *update*-запросом.

Как и в случае без дедубликации, выведем зависимость количества запросов к БД от размера блока при полном сохранении всех данных. Пусть n – количество файлов в текущей директории, m – количество папок в текущей директории, а sgm – длина блока. Число блоков файла длины l равно:

$$s = floor\left(\frac{l + sgm - 1}{sgm}\right),$$

где $floor(x)$ – функция нахождения ближайшего целого, не превышающего x. Тогда количество запросов к БД при сохранении содержимого текущей директории можно выразить следующей формулой:

$$F(m, n) = \sum_{i=1}^{n} (2s_i + \epsilon_i + 1) + m + 1,$$

где $sgm < l_i$, l_i – длина i-го файла в текущей директории, s_i – число блоков i-го файла, а $0 \leq \epsilon_i \leq s_i$ – количество сохраненных блоков.

Для каждого файла выполняется s_i *select*-запросов для определения наличия блока с заданным хешем в таблице *Data* и столько же *insert*-запросов в таблицу *Link*, ϵ_i *insert*-запросов для записи блоков в таблицу *Data*, один запрос для записи информации о файле в таблицу *Model*, m запросов для записи информации о папках в таблицу *Model* и еще один запрос для обновления записи в таблице *Model* (сохранение хеша текущей директории). ϵ будет равен нулю в случае, если директория содержит только копии уже сохраненных файлов, и примет значение s_i, если в директории отсутствуют копии сохраненных файлов.

Количество запросов к БД при сохранении содержимого директории и всех ее вложенных папок можно представить следующей формулой:

$$F_k = \sum_{i=1}^{n_k} (2s_{k_i} + \epsilon_{k_i}) + \sum_{j=1}^{m_k} F(m_{k_j}, n_{k_j}) + m + 1,$$

где k – идентификатор текущей директории.

В случае сохранения данных без дедубликации, формула имеет вид:

$$F_k = \sum_{i=1}^{n_k} (s_{k_i}) + \sum_{j=1}^{m_k} F(m_{k_j}, n_{k_j}) + m + 1.$$

Таким образом, за счет дедубликации, скорость сохранения уменьшится из-за дополнительных $\sum_{i=1}^{n_k} (s_{k_i} + \epsilon_{k_i})$ запросов.

Алгоритм восстановления данных остается таким же, как и в случае без дедубликации [1]. Меняется только текст запроса, который возвращает содержимое файла. Соответственно дедубликация практически не влияет на скорость восстановления данных.

1.2. Тестирование

Посмотрим, как это работает на практике. Тестовый стенд представляет собой два персональных компьютера (ПК), объединенных в сеть. Один из них играет роль сервера с установленной СУБД (процессор: Intel Core i5-2500K 3,3 GHz L3 6МБ, память 8 GB). Второй ПК представляет собой клиентскую часть системы, которая содержит пользовательские данные (процессор: Intel Celeron 2,8 GHz L2 256КБ, память: 3 GB). Жесткие диски обоих ПК отформатированы под файловую систему NTFS(3,15). Размер кластера составляет 4 КБ.

Система тестируется с применением СУБД Linter 6.0.18.9 Demo и MSSQL 2008 R2 с настройками по умолчанию [13,14].

СУБД MSSQL содержит компонент управления буфером, состоящий из двух механизмов: диспетчер буферов для доступа и обновления страниц базы данных, а также буферный кэш (известный как буферный пул) для сокращения операций ввода-вывода файла базы данных. На тестовом стенде заданы следующие настройки буферного кэша: количество доступной физической памяти (bpool_commit_target) 530358 КБ, объем физической памяти в диспетчере (bpool_committed) 8945 КБ.

СУБД Linter для своей работы использует так называемые очереди таблиц, файлов, колонок таблиц и пользователей. Размеры очередей задаются количеством элементов, размеры которых в разных очередях колеблются в среднем от 50 до 1500 байт, согласно документации. Тестовые данные были получены при следующих настройках: кэш файлов – 20 элементов, кэш таблиц – 100 элементов, кэш колонок – 500 элементов, кэш каналов – 100 элементов.

На стороне клиента запускается приложение, которое обрабатывает данные пользователя и отправляет их на сервер без предварительного шифрования. После того как все данные успешно сохранены, клиентское приложение запрашивает их обратно и сохраняет их на стороне клиента в другой директории. В качестве тестовых

данных были выбраны файлы разного размера. Для первых двух тестов было отобрано 76 файлов размером 0,6 МБ и столько же файлов по 3 МБ. В трёх других тестах происходит сохранение одного файла размером 22 МБ, 162 МБ и 1,36 ГБ. Для каждого теста все файлы расположены в одной директории без вложенных папок.

Наша задача заключается в выборе оптимального размера блока для файлов разного размера при включенном механизме исключения дублирования данных. Скорость обработки данных при использовании разных СУБД сильно отличается, однако характер влияния дедубликации на скорость сохранения в обоих случаях идентичен. Чем больше размер блока, тем меньше дедубликация влияет на скорость сохранения данных. С дедубликацией, при увеличении размера блока, скорость работы сохранения алгоритма растет быстрее до определенного момента.

Минимальный размер блока, 4 КБ, равен размеру кластера жесткого диска, отформатированного под файловую систему NTFS, а максимальный – 1 МБ. Выбор максимального размера блока связан с тем, что объекты размером 1 МБ и более эффективнее хранить в файловой системе, чем в БД, поскольку NTFS лучше справляется с фрагментацией таких объектов. Для хранения данных размером 256–1024 КБ ни файловая система, ни БД не дают явного преимущества. Объекты менее 256 КБ эффективнее хранить в БД [5].

Рассмотрим результаты первого теста (рисунок 2). Для СУБД MSSQL скорость сохранения данных растет при увеличении длины блока до 16 КБ, а потом не изменяется. Скорость восстановления начинает падать сразу с 4 КБ. Для СУБД Linter при длине блока более 32 КБ скорость сохранения данных растет медленнее, а скорость восстановления начинает увеличиваться. Таким образом, оптимальный размер блока составляет 32 КБ для Linter и 16 КБ для MSSQL. При этом файлы поделятся примерно на 19 и 38 частей соответственно.

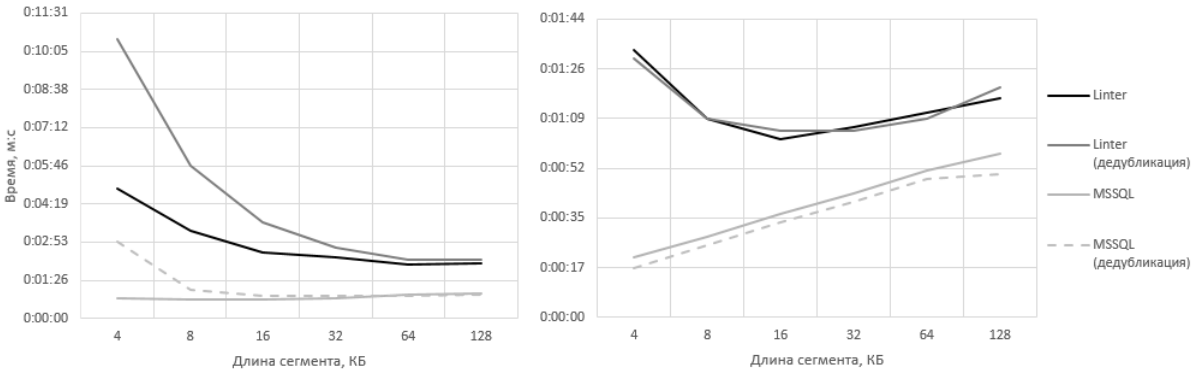


Рис. 2: Сохранение (слева) и восстановление (справа) 76 файлов по 0,6 МБ

Fig. 2. The saving (left) and recovery (right) of 76 files by 0,6 MB

Рассмотрим результаты второго теста (рисунок 3). Для СУБД MSSQL при увеличении длины блока более 16 КБ скорость сохранения данных растет медленнее, а скорость восстановления, так же как и в первом тесте, начинает падать сразу с 4 КБ. Для СУБД Linter увеличение блока больше 64 КБ не влияет на скорость восстановления, а скорость сохранения растет медленнее. Таким образом, оптимальный

размер блока может составлять 64 КБ для Linter и 16 КБ для MSSQL. При этом файлы будут разбиты примерно на 47 и 187 частей соответственно.

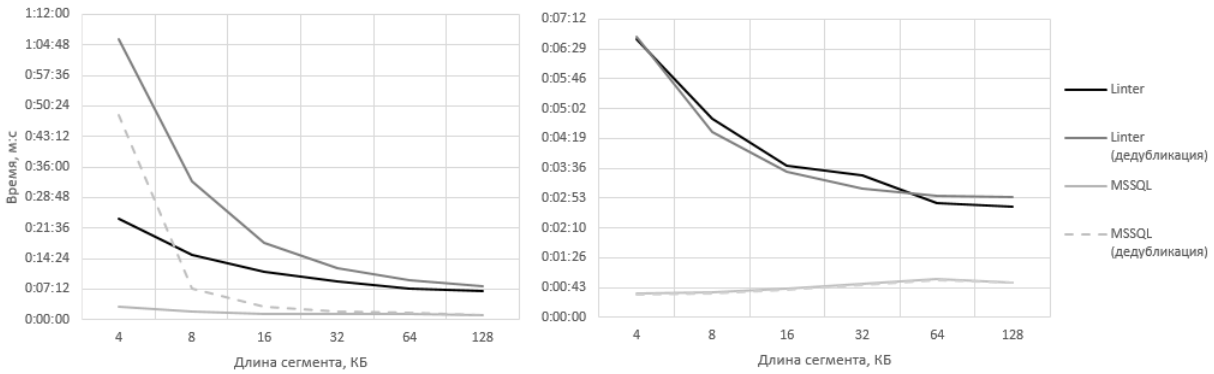


Рис. 3: Сохранение (слева) и восстановление (справа) 76 файлов по 3 МБ

Fig. 3. The saving (left) and recovery (right) of 76 files by 3 MB

Для СУБД MSSQL, при сохранении файла размером 22 МБ (рисунок 4), скорость сохранения растет быстро, а скорость восстановления не меняется при увеличении длины блока. Для Linter при увеличении блока больше 64 КБ скорость сохранения растет медленно, а скорость восстановления не меняется. Таким образом, оптимальный размер блока может составлять 64 КБ для Linter и 128 КБ для MSSQL. При этом файл будет разбит примерно на 344 и 688 частей соответственно.

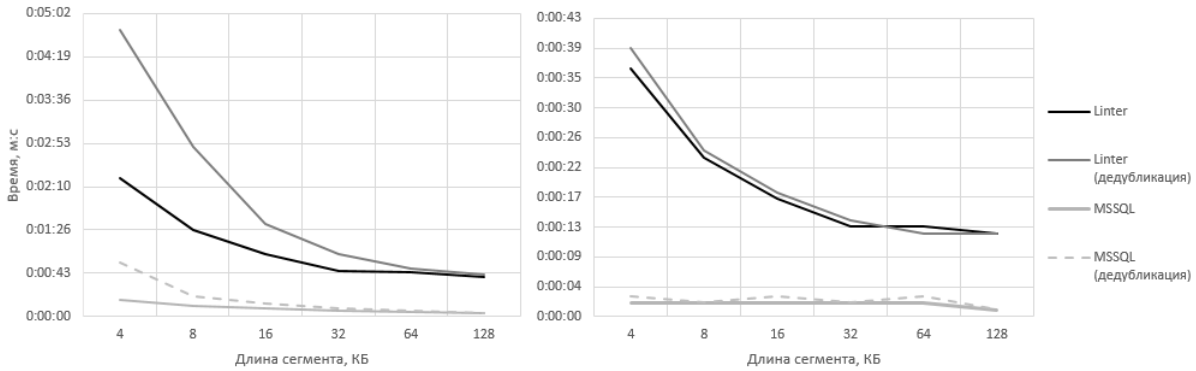


Рис. 4: Сохранение (слева) и восстановление (справа) файла, 22 МБ

Fig. 4. The saving (left) and recovery (right) of file, 22 MB

Для СУБД MSSQL, при сохранении файла размером 162 МБ (рисунок 5), скорость сохранения и восстановления растет быстро и равномерно для блоков больше 64 КБ. Для Linter увеличение длины блока более 256 КБ незначительно влияет на скорость сохранения данных, а увеличение более чем на 128 КБ незначительно влияет на скорость восстановления. Таким образом, оптимальный размер блока для Linter может составить 256 КБ, а для MSSQL – 1 МБ. При этом файл будет разбит примерно на 633 и 158 частей соответственно.

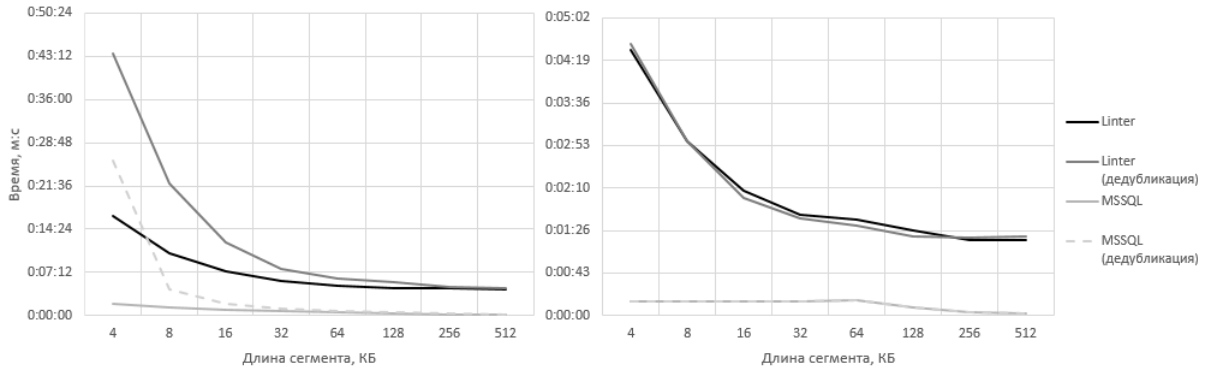


Рис. 5: Сохранение данных (1 файл, 162 МБ)

Fig. 5. The data saving (1 file, 162 MB)

В последнем тесте (рисунок 6) для СУБД Linter характер зависимости не изменился. Для MSSQL увеличение длины блока более 1 МБ незначительно влияет на скорость сохранения данных, а скорость восстановления быстро растет для блоков, длина которых больше 64 КБ. Однако для более эффективного хранения необходимо оставить длину блока на уровне 1 МБ. Таким образом, оптимальный размер блока может составлять 256 КБ для Linter и 1 МБ для MSSQL. При этом файл будет разбит примерно на 5312 и 1328 частей соответственно.

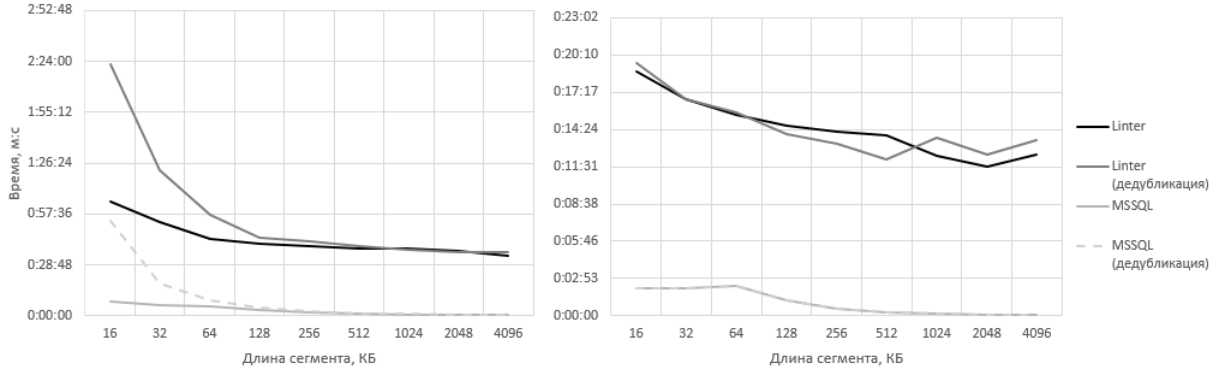


Рис. 6: Сохранение данных (1 файл, 1,36 ГБ)

Fig. 6. The data saving (1 file, 1,36 GB)

Таким образом, чтобы добиться наибольшей эффективности работы системы резервного копирования с дедубликацией, необходимо менять размер блока в зависимости от размера файла и СУБД, которая установлена на сервере (адаптивная дедубликация) [4]. Результаты тестов приведены в таблице 1.

Таблица 1. Результаты тестов

Table 1. The tests results

Тесты Tests		СУБД DBMS	
		MSSQL	Linter
Номер Number	Данные количество файлов / размер файла, МБ Data the files count / the size of file, MB	Оптимальный размер блока, КБ The optimal size of block, KB	
1	70 / 0.6	16	32
2	76 / 3	16	64
3	1 / 22	128	64
4	1 / 162	1024	256
5	1 / 1360	1024	256

Заключение

В статье был предложен подход к реализации механизма дедубликации в системе резервного копирования с хранением информации в базе данных. Архитектура такой системы позволяет внедрить данную технологию без существенного изменения программного кода. В зависимости от сетевой инфраструктуры, в которую устанавливается данная система, механизм исключения дублирования данных может быть выключен для достижения максимальной производительности. Если дедубликация необходима, то для повышения эффективности работы системы следует менять мелкость разбиения файлов в зависимости от их размера, а также от СУБД, которая обрабатывает данные на стороне сервера. В статье были приведены результаты соответствующих тестов.

Направлением дальнейшей работы является исследование поведения клиентской и серверной части предлагаемой системы при изменении файлов на клиенте, а также сетевой нагрузки при этом.

Список литературы / References

[1] Таранин С. М., “Резервное копирование с хранением в базе данных”, *Моделирование и анализ информационных систем*, **23**:4 (2016), 479–491; [Taranin S. M., “Backup with Storage in a Database”, *Modeling and Analysis of Information Systems*, **23**:4 (2016), 479–491, (in Russian).]

[2] Казаков В. Г., Федосин С. А., “Технологии и алгоритмы резервного копирования”, *Всероссийский конкурсный отбор обзорно-аналитических статей по приоритетному направлению «Информационно-телекоммуникационные системы»*, 2008, 1–49; [Kazakov V. G., Fedosin S. A., “Technologii i algoritmi reservnogo kopirovania”, *Vserossiyskiy konkursniy otbor obzorno-analiticheskih statey po prioritetnomu napravleniu "Informacionno-telekommunikacionnie sistemi"*, 2008, 1–49, (in Russian).]

[3] Medeiros J., “NTFS Forensics: A Programmers View of Raw Filesystem Data Extraction”, *Grayscale Research*, 2008, 1–27.

[4] Казаков В. Г., Федосин С. А., Плотникова Н. П., “Способ адаптивной дедупликации с применением многоуровневого индекса размещения копируемых блоков данных”, *Фундаментальные исследования*, 2013, № 8, 1322–1325; [Kazakov V. G., Fedosin S. A., Plotnikova N. P., “Method of adaptive dedublication with multilevel block indexing”, *Fundamental research*, 2013, № 8, 1322–1325].

[5] Sears R., Catharine van Ingen, Gray J., To BLOB or Not To BLOB: Large Object Storage in a Database or a Filesystem? *Technical Report MSR-TR-2006-45*, 2006, 1–11.

[6] Zhu N., Chiueh T., “Portable and Efficient Continuous Data Protection for Network File Servers”, *Stony Brook University*, 2007, 1–17.

[7] Meyer D. T., Bolosky W. J., “A Study of Practical Deduplication”, *ACM Transactions on Storage*, **7**:4 (2012), 1–13.

[8] Storer M. W., Greenan K., Long D. D. E., Miller E. L., “Secure Data Deduplication”, *Proceedings of the 4th ACM international workshop on Storage security and survivability*, 2008, 1–10.

[9] Renzel K., Keller W., “Client/Server Architectures for Business Information Systems”, *A Pattern Language*, 1997, 1–25.

[10] Дейт К. Дж., *Введение в системы баз данных*, **8**, Вильямс, 2005; In English: Date C. J., *An Introduction to Database Systems*, **8**, Pearson Education, Inc., 2004.

[11] Грофф Д., Вайнберг П., Оппель Э., *SQL: полное руководство*, **3**, Вильямс, 2015; In English: Groff J., Weinberg P., Oppel A., *SQL The Complete Reference*, **3**, The McGraw-Hill Companies, 2010.

[12] Дейт К. Дж., *SQL и реляционная теория. Как грамотно писать код на SQL*, Символ-Плюс, 2010; In English: Date C. J., *SQL and Relational Theory. How to Write Accurate SQL Code*, O’Reilly Media Inc., 2009.

[13] Mistry R., Misner S., *Introducing Microsoft SQL Server 2008 R2*, Microsoft Press, 2010.

[14] Максимов В., Козленко Л. А., Маркин С. П., Бойченко И. А., “Защищенная реляционная СУБД Линтер”, *Открытые системы. СУБД*, 1999, № 11–12; [Maksimov V., Kozlenko L. A., Markin C. P., Bojchenko I. A., “Zashchishchennaya relyacionnaya SUBD Linter”, *Otkrytye sistemy. SUBD*, 1999, № 11–12, (in Russian).]

[15] Таненбаум Э., Бос Х., *Современные операционные системы*, **4**, Питер, 2015; In English: Tanenbaum A. S., Bos H., *Modern Operating Systems*, **4**, Pearson Education, Inc., 2015.

Taranin S. M., "De-duplication on the Backup System with Information Storage in a Database", *Modeling and Analysis of Information Systems*, **24**:2 (2017), 215–226.

DOI: 10.18255/1818-1015-2017-2-215-226

Abstract. Prevention of data loss from digital media includes such a process as a backup. It can be done manually by copying data to external media or automated on a schedule by using special software. There are the remote backup systems, when data are saved over the network to the remote repository. Such systems are multi-user and they process large amounts of data. Shared storage can meet files containing the same fragments. The elimination of repeated data is based on the mechanism of de-duplication. It is a method of information compression, when the search of copies is performed in the entire dataset rather than within a single file. The main advantage of using this technology is a significant saving of disk space. However, the mechanism of eliminating repetitive data can significantly reduce the speed of saving and restoring information. This article is devoted to the problem of implementing such a mechanism in the backup system with information storage in a relational database. In this paper we consider an example of implementation of such a system working in two modes: with the de-duplication of data and without it. The article illustrates a class diagram for the development of a client part of application as well as the description of tables and relationships between them in a database that belongs to the backend. The author offers an algorithm of saving data wiht de-duplication, and also

gives the results of comparative tests on the speed of the algorithms of saving and restoring information when working with relational database management systems from different manufacturers.

Keywords: file, data, backup, de-duplication, database

About the authors:
Sergey M. Taranin, orcid.org/0000-0001-8117-7358, PhD,
P.G. Demidov Yaroslavl State University,
14 Sovetskaya str., Yaroslavl 150003, Russia,
e-mail: staranin0208@yandex.ru

Моделирование и анализ информационных систем. Т. 24, № 2 (2017), с. 227–238
Modeling and Analysis of Information Systems. Vol. 24, No 2 (2017), pp. 227–238

DOI: 10.18255/1818-1015-2017-2-227-238

УДК 62-50

Задачи оптимизации с усреднением по части переменных и условия их оптимальности в форме принципа максимума

Цирлин А. М.

получена 30 августа 2016

Аннотация. Рассмотрены задачи нелинейного программирования, критерий и ограничения которых усредненно зависят от части переменных. Показано, что если в этих задачах существует решение, то функция Лагранжа на нем достигает максимума по тем переменным, по которым происходит усреднение. При этом функции, определяющие задачу, могут быть не дифференцируемыми, а непрерывными по этим переменным, множество их допустимых значений может содержать и изолированные точки. В вариационных задачах может отсутствовать решение в классе кусочно-непрерывных функций по части переменных, но существовать обобщенное решение, на котором эти переменные изменяются в скользящем режиме, а критерий оптимальности стремится к своей верхней грани. Если же в таких задачах решение в классе кусочно – непрерывных функций существует, то условия оптимальности этого решения имеют форму принципа максимума функции Гамильтона. Рассмотрена связь усреднения по времени и по множеству значений переменных.

Ключевые слова: Усредненная оптимизация, расширение множества допустимых, эквивалентность расширения, вариация вероятностной меры, условия в форме принципа максимума

Для цитирования: Цирлин А. М., "Задачи оптимизации с усреднением по части переменных и условия их оптимальности в форме принципа максимума", *Моделирование и анализ информационных систем*, **24**:2 (2017), 227–238.

Об авторах:
Цирлин Анатолий Михайлович, orcid.org/0000-0002-3637-6160, д-р техн. наук, профессор,
Институт программных систем им. А.К. Айламазяна РАН
ул. Петра Первого, 4а, с. Веськово, Переславский р-он, Ярославская обл., 152020 Россия,
e-mail: tsirlin@sarc.botik.ru

Введение

Ниже будут рассмотрены экстремальные задачи, у которых критерий оптимальности I и ограничения усредненно зависят от части переменных. Такие задачи возникают объективно, когда в технологических процессах некоторые подлежащие выбору переменные должны быть неизменны (конструктивные параметры), а другие – могут изменяться во времени, причем наличие емкостей приводит к эффекту усреднения влияния этих изменений [1].

Они возникают также как вспомогательные, оценочные, когда введение усреднения расширяет множество допустимых решений и упрощает решение. Значение такой расширенной задачи заведомо «не хуже», чем значение исходной. А оптимальное ее решение содержит полезную информацию о характере оптимального

решения исходной. Для определенности будем рассматривать задачи на максимум критерия оптимальности.

Для некоторых классов задач удается доказать, что точные верхние грани критериев оптимальности исходной и расширенной задач совпадают. Расширение в этом случае называют *эквивалентным*. В частности, усредненное расширение вариационных задач эквивалентно по тем переменным, сколь угодно быстрые изменения которых „сглаживаются“ условиями задачи. Для эквивалентного расширения найдется одна или несколько последовательностей допустимых решений исходной задачи, на которых ее критерий оптимальности стремится к своей верхней грани (такое решение называют обобщенным), равной максимуму критерия расширенной, либо, если решение исходной задачи существует, оно совпадает с решением расширенной.

Первоначально рассмотрена связь между усреднением по времени и по множеству. Затем задача нелинейного программирования (НП) с усреднением по части переменных, наконец, вариационные задачи. Показано, что между условиями оптимальности этих задач есть много общего и возможность применения принципа максимума в вариационных задачах по некоторым переменным связана с эквивалентностью усредненного расширения по этим переменным.

1. Усреднение по времени и по множеству

1.1. Время не входит явно в условия задачи

Среднее значение функции $f(u(t))$ на интервале $(0,\tau)$ может быть вычислено по времени как

$$\overline{f_t(u)} = \frac{1}{\tau}\int\limits_0^\tau f(u(t))dt \tag{1}$$

или по множеству, если для любого t $u \in V \in R^m$

$$\overline{f_p(u)} = \int\limits_V f(u)p(u)du, \tag{2}$$

где $p(u)$ — вероятностная мера

$$p(u) \geq 0, \quad \int\limits_V p(u)du = 1, \tag{3}$$

для любого $V_0 \in V$, величина

$$S_0 = \int\limits_{V_0} p(u)du$$

равна доле интервала $(0,\tau)$, в течение которой значения $u(t)$ принадлежат V_0.

Кусочно-постоянной функции $u(t)$, принимающей дискретные значения $u_i \in V_u$, соответствует вероятностная мера вида

$$p(u) = \sum_i \gamma_i \delta(u-u_i), \quad \gamma_i \geq 0, \quad \sum \gamma_i = 1. \tag{4}$$

Интеграл (2) в этом случае примет вид

$$\overline{f_p(u)} = \sum_i \gamma_i f(u^i). \tag{5}$$

На величину интегралов в (1), (2) не влияет последовательность, в которой $u(t)$ принимает значения u^i. Таким образом, каждой вероятностной мере $p(u)$ может соответствовать много функций $u(t)$, для которых $\overline{f_t(u)} = \overline{f_p(u)}$. Если $p(u) = \delta(u - u^0)$, то $u(t) = u_0 = const$. Если число базовых значений больше единицы, то число функций $u(t)$ сколь угодно велико, т.к. каждое из значений u^i она может принимать неоднократно, лишь бы суммарная доля интервала $[0, \tau]$, в течение которой $u(t) = u^i$, составляла γ_i.

Пусть ставится задача о максимуме среднего значения функции $f_0(u)$ на V. Если эта функция унимодальна и достигает максимума в точке u^0, то решение единственно

$$u^*(t) = u^0, \quad p^*(u) = \delta(u - u^0), \tag{6}$$

но если функция $f_0(u)$ достигает максимума в r точках u^i, то решений $u^*(t)$, доставляющих максимум ее среднего значения, сколь угодно много, они кусочно-постоянные и всем им соответствует одна и та же вероятностная мера

$$p^*(u) = \sum_{i=1}^{r} \gamma_i \delta(u - u^i), \tag{7}$$

причем для всех γ_i, удовлетворяющих условиям (4), значение $\overline{f^*(u)}$ одинаково. Будем называть значения u^i — базовыми.

Появляется возможность, не „ухудшая“ $\overline{f_0^*(u)}$, наложить дополнительные усредненные ограничения и выполнить их за счет выбора γ_i. Число m таких ограничений не более чем $r-1$, так как переменные γ_i неотрицательны и в сумме равны единице.

Таким образом, чтобы было выполнено m усредненных условий вида

$$\overline{f_\nu(u)} = \sum_{i=1}^{r} \gamma_i f_\nu(u^i) = 0, \quad \nu = \overline{1, m}, \tag{8}$$

функция $f_0(u)$ должна иметь не менее чем $(m+1)$ максимум.

1.2. Усреднение функций, явно зависящих от времени

Пусть $f = f(t, u(t))$ непрерывна по совокупности переменных. Ее среднее значение по времени

$$\overline{f_t(t,u)} = \frac{1}{\tau} \int_0^\tau f(t, u(t))dt, \quad u \in V(t). \tag{9}$$

Любой кусочно-непрерывной функции $u^0(t)$ соответствует вероятностная мера $p^0(u,t)$

$$p^0(u,t) = \delta(u - u^0(t)), \tag{10}$$

такая, что

$$\overline{f_p(u^0)} = \frac{1}{\tau}\int\limits_0^\tau \int\limits_{V(t)} f(t,u)p^0(u,t)du\,dt. \tag{11}$$

В более общем случае вероятностная мера $p(u,t)$ не сосредоточена при каждом t в единственной точке u^0, как $p^0(u,t)$, а удовлетворяет только требованиям

$$\int\limits_{V(t)} p(u,t)du = 1, \quad p(u,t) \geq 0 \quad \forall t \in [0,\tau]. \tag{12}$$

Расширенное множество вероятностных мер $p(u,t)$ включает $p^0(u,t)$. Для любой $p(u,t)$ найдется такая последовательность функций $\{u^\nu(t)\}$, на которой

$$\lim_{\nu\to\infty} \overline{f_t(t,u^\nu)} = \frac{1}{\tau}\int\limits_0^\tau \int\limits_{V(t)} f(t,u)p(u,t)du\,dt. \tag{13}$$

Следуя Л. Янгу [2], будем называть меру $p(u,t)$ *обобщенной переменной.*

2. Задача НП с усреднением по некоторым переменным

Рассмотрим задачу

$$\overline{f_0(x,u)}^u \to \max \Big/ \begin{array}{ll} \overline{f_i(x,u)}^u = 0, & i = \overline{1,m}, \\ u \in V \subset R^k, & x \in R^{n-k}. \end{array} \tag{14}$$

Здесь черта над функциями f_0, f_i соответствует их усреднению по u:

$$\overline{f_i(x,u)}^u = \int\limits_V f_i(x,u)p(u)du, \quad i = \overline{0,m}. \tag{15}$$

В задаче (14) два типа переменных: вектор x и вероятностная мера $p(u)$, удовлетворяющая условиям (3). Функции f_i непрерывны по x,u и непрерывно дифференцируемы по x, множество V – компакт. Для простоты не рассматриваем задачу с неравенствами, т.к. их наличие ничего в существе дела не меняет.

Зафиксируем $x \in R^{n-k}$ и рассмотрим семейство вспомогательных задач

$$f_0(x,u) \to \max_u \Big/ \begin{array}{ll} f_i(x,u) = c_i, & i = \overline{1,m}, \\ u \in V \subset R^k. \end{array} \tag{16}$$

Обозначим оптимальное решение этой задачи через $u^*(x,c)$, а значение через $f_0^*(x,c) = f_0(x,u^*(x,c))$. Функцию $f_0^*(x,c)$ называют функцией достижимости [5]. Отметим, что для некоторых c задача (16) может не иметь решения (множество ее допустимых решений пусто). Множество тех c, для которых задача имеет решение,

обозначим через $V_c \in R^m$. Очевидно, что для любого x оптимальная мера $p(u)$ в задаче (14), (15) сосредоточена только на решениях $u^*(x,c)$ задачи (16).

С использованием введенных обозначений задачу (14) можно переписать в форме

$$\overline{f_0(x,c)}^c \to \max_{p(x,c)} \Big/ \begin{array}{l} \overline{c_i} = 0, \quad i = \overline{1,m}, \\ c \in V_c(x) \in R^m. \end{array} \tag{17}$$

Но это — задача об ординате выпуклой оболочки функции $f_0(x,c)$ в точке $c(x) = 0$.

Согласно теореме Каратеодори [3] оптимальное решение такой задачи имеет вид

$$p^*(x,c) = \sum_{j=0}^{m} \gamma_j(x)\delta(c - c^j(x)), \quad \gamma_j(x) \geq 0, \quad \sum_{i=0}^{m} \gamma_j(x) = 1,$$

т.е. оно сосредоточено не более чем в $(m+1)$ точке независимо от размерности u, x. При этом каждому из значений $c^j(x)$ соответствует вектор $u^j(x)$.

Этот факт позволяет переписать задачу (14) как задачу нелинейного программирования (НП)

$$\sum_{j=0}^{m} \gamma_j f_0(x,u^j) \to \max \Big/ \begin{array}{l} \sum\limits_{j=0}^{m} \gamma_j f_i(x,u^j) = 0, \quad i = \overline{1,m}, \\ u^j \in V \subset R^k, \quad x \in R^{n-k}, \\ \sum\limits_{j=0}^{m} \gamma_j = 1, \quad \gamma_j \geq 0, \end{array} \tag{18}$$

переменными в которой являются вектор x, вектор весовых коэффициентов γ и векторы u^j.

Задаче (18) соответствует функция Лагранжа

$$R = \sum_{j=0}^{m} \gamma_j \left[\sum_{i=0}^{m} \lambda_i f_i(x,u^j) - \Lambda \right], \tag{19}$$

в которой $\lambda_0 = 1$ или 0.

С использованием функции R могут быть сформулированы условия оптимальности решения $z^* = (x^*, u^{j*}, \gamma_j^*)$ задачи (18).

Достаточное: *Для того чтобы решение z^* было оптимальным, достаточно, чтобы нашелся такой вектор множителей Лагранжа $\lambda^* = (1, \lambda_1^*, ..., \lambda_m^*, \Lambda^*)$, чтобы функция R достигала максимума и были выполнены ограничения задачи* (18). Отметим, что число таких максимумов не превышает $(m+1)$ и равно числу базовых значений u^j.

Доказательство этих условий очевидно, но существование такого вектора не гарантировано.

Необходимое:

Если z^* — оптимальное решение, то найдется такой ненулевой вектор $\lambda^* = (\lambda_0, ..., \Lambda)$, $\lambda_0 = (0;1)$, что

$$\frac{\partial R}{\partial x} = \frac{\partial}{\partial x} \sum_{j=0}^{m} \gamma_i L(x,u^j,\lambda^*) = 0, \tag{20}$$

$$u^{j*} = \arg\max_{u\in V} L(x,u,\lambda^*), \quad j=\overline{0,m}, \quad \gamma_j \geq 0, \quad \sum_{j=0}^{m}\gamma_j = 1. \tag{21}$$

Здесь

$$L(x,u,\lambda) = \sum_{i=0}^{m}\lambda_i f_i(x,u)$$

функция Лагранжа неусредненной задачи.

Таким образом, по усредняемым переменным и функция L достигает максимума на оптимальном решении, а ее среднее по и значение стационарно по x.

Доказательство необходимых условий оптимальности следует из теоремы Куна – Таккера [4] применительно к задаче (18). По условию локальной неулучшаемости R по весовым коэффициентам γ_j с учетом их неотрицательности получим выражения

$$\frac{\partial R}{\partial \gamma_j}\delta\gamma_j \leq 0 \Rightarrow \begin{cases} \frac{\partial R}{\partial \gamma_j} \leq 0 & \text{для} \quad \gamma_j = 0, \\ \frac{\partial R}{\partial \gamma_j} = 0 & \text{для} \quad \gamma_j > 0. \end{cases} \tag{22}$$

А так как $\frac{\partial R}{\partial \gamma_j} = L(x,u^j,\lambda) - \Lambda$, то из (22) вытекает, что с положительными весами в функцию R на оптимальном решении входят только такие значения u^j (базовые), на которых L максимальна по u на V и равна Λ для всех j.

Это обстоятельство позволяет ослабить требования к условиям оптимальности задачи: не требовать гладкости функций f_i по u; множество V может состоять из изолированных значений u и пр.

3. Вариационные задачи

Рассмотрим задачи, в которых искомыми решениями являются функции.

3.1. Задача с интегральными ограничениями

$$I = \int_0^{\tau} f_0(x,u(t),t)dt \to \max, \tag{23}$$

$$J_i = \int_0^{0} f_i(x,u(t),t)dt = 0, \quad i=\overline{1,m}, \quad u(t)\in V(t), \quad x\in R^n.$$

В этой задаче функции $f_i (i=\overline{0,m})$ непрерывны по u и непрерывно-дифференцируемы по $x,t,V(t)$ — компакт, $u(t)$ — кусочно-непрерывная.

Сопоставим задаче (23) усредненную по u задачу (см. [5], [6])

$$\overline{I} = \int_0^{\tau}\int_{V(t)} f_0(x,u,t)p(u,t)du\,dt \to \max, \tag{24}$$

$$\overline{J}_i = \int_0^{\tau}\int_{V(t)} f_i(x,u,t)p(u,t)du\,dt = 0, \quad i=\overline{1,m},$$

$$\int\limits_{V(t)} p(u,t)du = 1, \quad p(u,t) \geq 0, \quad \forall t \in [0,\tau].$$

Каждой функции $u_0(t)$ в задаче (23) можно сопоставить меру

$$p_0(u,t) = \delta(u - u_0(t)). \tag{25}$$

При этом $\overline{I} = I$, а $J_i = \overline{J_j}$. В классе решений, имеющих вид (25), задачи (23) и (24) отличаются только формой записи, а по существу они одинаковы. Между множествами допустимых решений задачи (23) и задачи (24) в форме (25) существует взаимнооднозначное соответствие, такое, что на соответствующих друг другу решениях значения критерия и ограничений одинаковы. Такие задачи называют изоморфными [5], [7].

Множество допустимых решений задачи (24) шире, чем множество решений вида (25). Так что задача (24) является расширением для (23) и

$$\overline{I}^* \geq I^*. \tag{26}$$

Однако каждому решению $p_0(u,t)$ задачи (24) можно сопоставить последовательность решений $\{u_\nu(t)\}$, на которой предел любого из ограничений J_j стремится к нулю, а предел I стремится к $\overline{I}(x^*, p_0(u,t))$. Так что, если под I^* понимать точную верхнюю грань I на множестве допустимых решений, то неравенство (26) можно заменить равенством. Таким образом, расширение (24) задачи (23) эквивалентно.

Справедлива следующая Л е м м а: *Пусть расширение исходной экстремальной задачи эквивалентно. Тогда необходимые условия оптимальности решения расширенной задачи являются необходимыми условиями оптимальности решения исходной, если оно существует.*

Действительно, необходимые условия оптимальности решения задачи (24) совпадают с необходимыми условиями оптимальности задачи (23) в классе обобщенных решений. В том случае, если в задаче (23) существует оптимальное решение ($\mathrm{Sup} I = \max I$), то необходимые условия оптимальности задачи (24) выделяют решение вида (25), а значит, и оптимальное решение задачи (23).

Сформулируем эти условия: *Пусть* $x^*, p^*(u,t)$ *– оптимальное решение задачи* (24), *тогда существует такой ненулевой вектор* $\lambda = (\lambda_0, ..., \lambda_m), \lambda_0 = (0;1)$, *что функционал*

$$S = \lambda_0 \overline{I} + \sum_{i=1}^{m} \lambda_i \overline{J_i} \tag{27}$$

стационарен по x, *а функция*

$$L = \sum_{i=0}^{m} \lambda_i f_i(x,u,t) \tag{28}$$

достигает максимума по $u \in V(t)$:

$$\int\limits_0^\tau \frac{\partial}{\partial x} \sum_{i=0}^{m} \int\limits_{V(t)} \lambda_i f_i(x,u,t) p^*(u,t) du\, dt = 0, \tag{29}$$

$$u^*(t) = \arg \max_{u \in V(t)} \sum_{i=0}^{m} \lambda_i f_i(x^*, u, t). \tag{30}$$

Справедливость этих условий следует из того, что задача (24) по отношению к вектору x является задачей нелинейного программирования с функцией Лагранжа

$$\begin{gathered}\overline{S} = S - \int_0^\tau \Lambda(t) \int_{V(t)} p(u,t) du\, dt = \int_0^\tau \int_{V(t)} \left[\sum_{i=0}^{m} \lambda_i f_i(x,u,t) + \right. \\ \left. + \Lambda(t) \right] p(u,t) du\, dt = \int_0^\tau \int_{V(t)} [L - \Lambda(t)] p(u,t) du\, dt,\end{gathered} \tag{31}$$

а для x^* подынтегральное выражение функционала $\overline{S}$ должно быть локально неулучшаемо по $p(u,t)$

$$\frac{\partial}{\partial p}\{[L - \Lambda(t)]p(u,t)\}\delta p \le 0, \tag{32}$$

откуда

$$\left.\begin{array}{l} L(x^*, u^*, t) = \Lambda(t) \quad \text{при} \quad p(u,t) > 0, \\ L(x^*, u, t) \le \Lambda(t) \quad \text{при} \quad p(u,t) = 0. \end{array}\right\} \tag{33}$$

Таким образом, с ненулевой мерой в задачу войдут только те значения $u(t)$, на которых L максимальна по $u \in V(t)$.

Если этот максимум единственный, то оптимальная мера имеет вид (25) и соответствующая функция $u^*(t)$ представляет собой решение задачи (23), если на некотором множестве значений t ненулевой меры максимум L по u не единственный, то оптимальному решению задачи (24) соответствует обобщенное решение задачи (23), на котором I достигает своей верхней грани.

Подчеркнем, что эквивалентность расширения связана с тем, что $u(t)$ входит в условия задачи под знаком интеграла, а следовательно, сколь угодно быстрые изменения этой переменной усредняются.

3.2. Задача оптимального управления

В задаче оптимального управления переменные состояния x и управляющие воздействия u зависят от t

$$I = \int_0^\tau f_0(x(t), u(t), t) dt \to \max, \tag{34}$$

$x(t)$ и $u(t)$ связаны дифференциальными уравнениями, которые могут быть переписаны как

$$x_i(t) = x_i(0) + \int_0^t f_i(x(t_1), u(t_1), t_1) dt_1, \quad i = \overline{1,m}, \quad u \in V(t). \tag{35}$$

Функции $f_i (i = 0, m)$ непрерывны по u и непрерывно дифференцируемы по x, t; $V(t)$ — компакт.

Для записи расширенной задачи введем обозначения:

$$\overline{f_i(x,u,t)} = \int\limits_{V(t)} f_i(x,u,t)p(u,t)du, \quad i=\overline{0,m}. \tag{36}$$

Расширенная задача примет форму

$$\overline{I} = \int\limits_0^\tau \overline{f_0(x,u,t)}dt \to \max_{x,p} \tag{37}$$

при условиях

$$\left.\begin{array}{l} \overline{J_i}(t) = \int\limits_0^\tau \left\{[x_i(t_1)-x_i(0)]\delta(t_1-t) - h(t-t_1)\overline{f_i(x,u,t_1)}\right\} dt_1 = 0, \\ J_{m+1}(t) = \int\limits_{V(t)} p(u,t)du - 1 = 0, \quad \forall t\ p(u,t) \geq 0, \quad i=\overline{1,m}. \end{array}\right\} \tag{38}$$

Искомыми переменными в этой задаче являются вектор-функция $x(t)$ и вероятностная мера $p(u,t)$.

Необходимые условия оптимальности расширенной задачи: *Пусть* $x^*(t), p^*(u,t)$ — *оптимальное решение задачи* (37), (38), *тогда найдется такая ненулевая вектор-функция*

$$\lambda(t) = (\lambda_0, \lambda_1(t), ..., \lambda_m(t), \Lambda(t)), \tag{39}$$

что на оптимальном решении

$$\frac{\partial}{\partial x_i}\left\{\lambda_0\overline{f_0(x,u,t)} + \sum_{i=1}^m \left[\lambda_i(t)x_i(t) - \int\limits_t^\tau \lambda_i(t_1)dt_1\overline{f_i(x,u,t)}\right]\right\} = 0, i=\overline{1,m}, \tag{40}$$

$p^*(u,t) > 0$, для

$$u^*(t) = \arg\max_{u\in V(t)}\left\{\lambda_0 f_0(x,u,t) + \sum_{i=1}^m \left[f_i(x,u,t)\cdot\int\limits_t^\tau \lambda_i(t)dt - \lambda_i(t)x_i(t)\right]\right\}. \tag{41}$$

Действительно, функционал Лагранжа расширенной задачи имеет вид

$$S = \lambda_0\overline{I} + \sum_{i=1}^m \int\limits_0^\tau \left(\lambda_i(t)\overline{J_i}(t) + \Lambda(t)J_{m+1}(t)\right)dt. \tag{42}$$

Во втором слагаемом S

$$-\int\limits_0^\tau \lambda_i(t)h(t-t_1)dt = -\int\limits_{t_1}^\tau \lambda_i(t)dt \quad i=\overline{1,m}. \tag{43}$$

С учетом (43) подынтегральное выражение в S можно записать как

$$\overline{R} = \lambda_0\overline{f_0(x,u,t)} + \sum_{i=1}^m \left[\lambda_i(t)x_i(t) - \int\limits_t^\tau \lambda_i(t_1)dt_1\cdot\overline{f_i(x,u,t)}\right] - \Lambda(t)p(u,t). \tag{44}$$

Введем обозначения:

$$\left.\begin{aligned} &\psi_i(t) = -\int\limits_t^\tau \lambda_i(t_1)dt_1 \to \dot{\psi}_i(t) = \lambda_i(t), \quad i = \overline{1,m}, \\ &\overline{H} = \lambda_0\overline{f_0(x,u,t)} + \sum_{i=1}^m \psi_i(t)\overline{f_i(x,u,t)}. \end{aligned}\right\} \tag{45}$$

Тогда необходимые условия оптимальности расширенной задачи (37), (38) примут форму

$$\left.\begin{aligned} &\tfrac{\partial\overline{R}}{\partial x_i} = \tfrac{\partial}{\partial x_i}\left[\overline{H} + \sum_{i=1}^m \dot{\psi}_i(t)x_i\right] = 0, i = \overline{1,m} \\ &u_\nu^*(t) = \arg\max\nolimits_{u\in V(t)} H(x,\psi,u,t). \end{aligned}\right\} \tag{46}$$

Если для каждого t максимум H по u единственный, то в силу непрерывности по u функций $f_i(x,u,t), i = 0,m$, решение исходной задачи существует и удовлетворяет условиям (46), совпадающим с принципом максимума Понтрягина. Если максимум не единственный, то исходная задача имеет обобщенное решение (скользящий режим), на котором достигается

$$\operatorname{Sup} I = \max \overline{I}.$$

Таким образом, возможность требования максимума функции Гамильтона по управляющим воздействиям связана с тем, что усредненное расширение задачи по этим переменным эквивалентно.

4. Заключительные замечания

Использованный здесь подход, названный в [8] „вариацией скольжения“, по существу сводится к замене исходной задачи ее усредненным расширением по той части переменных, для которых это расширение эквивалентно, т.е. любому решению усредненной задачи соответствует либо решение исходной, либо последовательность таких решений, на которой условия исходной задачи выполнены с любой наперед заданной точностью, а величина критерия исходной задачи сколь угодно близка к величине критерия усредненной. В этом случае при дополнительном предположении о существовании решения исходной задачи оно удовлетворяет условиям оптимальности расширенной. В противном случае эти условия выделяют оптимальный скользящий режим.

В [8] наличие смешанных ограничений на управляющие и фазовые переменные привело к необходимости разделять управления на два класса u_1 и u_2, по первому из которых допустимы слабые вариации.

Изложенный выше подход [9], в отличие от игольчатой вариации (см. [10]), пригоден для вариационных задач с сочетанием условий разного типа.

В задачах с ограничениями разного типа, а не только в форме дифференциальных уравнений, нет смысла выделять управляющие переменные и переменные состояния или делить управления на классы. Важно выделить те переменные, усредненное расширение задачи по которым эквивалентно. Для этого целесообразно условия задачи переписать в канонической интегральной форме [11], как это сделано вы-

ше для дифференциальных уравнений. Расширение задачи по некоторой переменной эквивалентно, если при такой записи она во все условия и в критерий оптимальности входит под знаком интеграла, а значит, ее сколь угодно быстрые изменения влияют на них усредненно.

Список литературы / References

[1] Цирлин А. М., *Оптимальные циклы и циклические режимы*, Энергоатомиздат, М., 1983; [Tsirlin A. M., *Optimalnie zikly i ziklisheskie regimi*, Energoatomizdat, M., 1983, (in Russian).]

[2] Янг Л., *Лекции по вариационному исчислению и теории оптимального управления*, Мир, М., 1977; [Jang L., *Lekzii po variazionnomu ishisleniju i teoriin optimalnogo upravlenija*, Mir, M., 1977, (in Russian).]

[3] Fromovitz St., "Non-linear programmingn with randomisation", *Manag. Sci. A.*, **11**:9 (1965).

[4] Himmelblau D. M., *Applied Nonlinear Programming*, N-Y, 1972.

[5] Цирлин А. М., *Методы усредненной оптимизации и их приложения*, Физматлит, М., 1997; [Tsirlin A. M., *Metodi usrednennoy optimizazii i ix prilogenija*, Fizmatkit, M., 1997, (in Russian).]

[6] Цирлин А. М., "Задачи и методы усредненной оптимизации", *Труды Математического института им. Стеклова*, **261** (2008), 1–17; [Tsirlin A. M., "Zadashi i metodi usrednennoy optimizazii", *Trudi instituta im. Steklova*, **261** (2008), 1–17, (in Russian).]

[7] Афанасьев А. П., Дикусар В. В., Милютин А. А., Чуканов С. А., *Необходимое условие в оптимальном управлении*, Наука, М., 1990; [Afanasyev A. P., Dicusar V. V., Milutin A. A., Shukanov C. A., *Neobxodimoe uslovie v optimalnom upravlenii*, Nauka, M., 1990, (in Russian).]

[8] Дубовицкий А. Я., Милютин А. А., "Теория принципа максимума", *Методы теории экстремальных задач в экономике*, Наука, М., 1981, 6–47; [Dubovizky A. J., Milutin A. A., "Teorija prinzipa maksimuma", *Metodi teorii ecstremalnix zadash v ekonomike*, Nauka, M., 1981, 6–47, (in Russian).]

[9] Цирлин А. М., "Оптимизация в среднем и скользящие режимы в задачах оптимального управления", *Изв. АН СССР. Техн. кибернетика*, 1974, № 2, 143–151; [Tsirlin A. M., "Optimizacija v srednem i skolzjashie regimi v zadashax optimalnogo upravlenija", *Izv. AN SSSR. Texn. kibernetika*, 1974, № 2, 143–151, (in Russian).]

[10] Розоноэр Л. И., "Принцип максимума Понтрягина в теории оптимальных систем", *Автомат. и телемех.*, **20**:10 (1959), 1320–1334; [English transl.: [Rozonoer L. I., "The Maximum Principle in the theory of optimal systems", *Autom. Remote Control*, **20**:10 (1959), 1320–1334].

[11] Цирлин А. М., "Условия оптимальности скользящих режимов и прнцип максимума для задачи со скалярным аргументом", *Автоматика и телемеханика*, 2009, № 5, 106–121; [English transl.: Tsirlin A. M., "Optimality conditions of sliding modes and the maximum principle for control problems with the scalar argument", *Autom. Remote Control*, **70**:5 (2009), 839–854].

Tsirlin A. M., "Optimization Problems with Averaging over the Variables", *Modeling and Analysis of Information Systems*, **24**:2 (2017), 227–238.

DOI: 10.18255/1818-1015-2017-2-227-238

Abstract. The problems of nonlinear programming, criteria and limitations depend on the variables averaged. It is shown that if these problems have solutions, the Lagrangian reaches the maximum

for the variables, which are averaged. The functions defining the problem can not be differentiable and continuous on these variables, the set of possible values may contain isolated points. In variational problems there can be no solution in the class of piecewise continuous functions of the variables, but there can be a generalized solution in which these variables change in the sliding mode, and the optimality criterion tends to its upper edge. If in such problems the solution in the class of piecewise - continuous functions exists, the conditions of optimality of this solution are in the form of the Hamiltonian function of the maximum principle. The relationship between the average over time and across multiple variables is considered.

Keywords: The average optimization, expansion of the set of admissible equivalence extension, variation of probability measures, the conditions in the form of the maximum principle

About the authors:
Anatoly M. Tsirlin, orcid.org/0000-0002-3637-6160, Prof,
Program Systems Institute of RAS
4a Petra 1 str., Veskovo Jaroslavskoy 152020, Russia, e-mail: tsirlin@sarc.botik.ru

Моделирование и анализ информационных систем. Т. 24, № 2 (2017), с. 239–252
Modeling and Analysis of Information Systems. Vol. 24, No 2 (2017), pp. 239–252

DOI: 10.18255/1818-1015-2017-2-239-252

УДК 517.9

Декодирование тензорного произведения MLD-кодов и приложения к кодовым криптосистемам

Деундяк В. М., Косолапов Ю. В., Лелюк Е. А.

получена 7 апреля 2017

Аннотация.

Для практического применения кодовой криптосистемы типа Мак-Элиса необходимо, чтобы используемый в основе криптосистемы код имел быстрый алгоритм декодирования. С другой стороны, используемый код должен быть таким, чтобы нахождение секретного ключа по известному открытому ключу было практически неосуществимо при относительно небольшом размере ключа. В связи с этим в настоящей работе предлагается в криптосистеме типа Мак-Элиса использовать тензорное произведение $C_1 \otimes C_2$ групповых MLD-кодов C_1 и C_2. Алгебраическая структура кода $C_1 \otimes C_2$ в общем случае отличается от структуры кодов C_1 и C_2, поэтому представляется возможным построение стойких криптосистем типа Мак-Элиса даже на основе кодов C_i, для которых известны успешные атаки на ключ. Однако на этом пути возникает проблема декодирования кода $C_1 \otimes C_2$. Основной результат настоящей работы – построение и обоснование набора необходимых для декодирования этого кода быстрых алгоритмов. Процесс построения декодера существенно опирается на групповые свойства кода $C_1 \otimes C_2$. В качестве приложения в работе построена криптосистема типа Мак-Элиса на коде $C_1 \otimes C_2$ и приводится оценка ее стойкости к атаке на ключ в предположении, что для кодовых криптосистем на кодах C_i возможна эффективная атака на ключ. Полученные результаты численно проиллюстрированы в случае, когда C_1, C_2 – коды Рида–Маллера–Бермана, для которых соответствующая кодовая криптосистема взломана Л. Миндером и А. Шокроллахи (2007 г.).

Ключевые слова: мажоритарный декодер, коды Рида–Маллера–Бермана, тензорное произведение кодов

Для цитирования: Деундяк В. М., Косолапов Ю. В., Лелюк Е. А., "Декодирование тензорного произведения MLD-кодов и приложения к кодовым криптосистемам", *Моделирование и анализ информационных систем*, **24**:2 (2017), 239–252.

Об авторах:
Деундяк Владимир Михайлович, orcid.org/0000-0001-8258-2419, канд. физ.-мат. наук, доцент,
ФГНУ НИИ "Спецвузавтоматика",
пер. Газетный, 51, г. Ростов-на-Дону, 344002 Россия,
Южный Федеральный Университет,
ул. Большая Садовая, 105/42, г. Ростов-на-Дону, 344006 Россия, e-mail: vl.deundyak@gmail.com,

Косолапов Юрий Владимирович, orcid.org/0000-0002-1491-524X, канд. техн. наук,
Южный Федеральный Университет,
ул. Большая Садовая, 105/42, г. Ростов-на-Дону, 344006 Россия, e-mail: itaim@mail.ru,

Лелюк Евгений Андреевич, orcid.org/0000-0001-6560-2561, магистрант,
Южный Федеральный Университет,
ул. Большая Садовая, 105/42, г. Ростов-на-Дону, 344006 Россия, e-mail: lelukevgeniy@mail.ru,

Введение

Стойкость применяемых в настоящее время на практике асимметричных криптосистем основана на сложности задач факторизации целых чисел или дискретного логарифмирования в конечной группе. Однако в [1] показано, что эти задачи могут быть решены за полиномиальное время на квантовом компьютере. Криптографические системы, в основе которых лежит применение помехоустойчивых кодов (далее — кодовые криптосистемы), рассматриваются в настоящее время как одна из альтернатив используемым в настоящее время асимметричным криптографическим системам [2]. Недостатком кодовых криптосистем является большой размер ключа. В частности, размер ключа для первой кодовой криптосистемы на основе кодов Гоппы, предложенной Робертом Мак-Элисом в [3], составляет порядка 65 Кбайт. Попытки уменьшить размер ключа за счет использования кодов, отличных от кодов Гоппы, не дали должного результата, так как предложенные системы оказались нестойкими. К нестойким относятся такие известные системы, как криптосистема Нидеррайтера [4], криптосистема Габидулина–Парамонова–Третьякова [5], криптосистема Сидельникова [6]. Для перечисленных криптосистем имеются эффективные структурные атаки, то есть атаки, направленные на нахождение подходящего секретного ключа по известному открытому ключу (см. [7]– [11]).

Представляется, что усилить стойкость кодовых криптосистем к структурным атакам возможно путем использования помехоустойчивых кодов, для которых, с одной стороны, имеется быстрый алгоритм декодирования, а с другой стороны, которые не обладают явно выраженной алгебраической структурой. Такой подход применен, например, в [12], где предлагается в криптосистеме типа Мак-Элиса использовать коды, индуцированные групповыми кодами. В настоящей работе приводится одно обобщение этого подхода: в качестве помехоустойчивого кода предлагается применять тензорное произведение $C_1 \otimes C_2$ двух групповых мажоритарно-декодируемых кодов C_1 и C_2 (MLD-кодов). Но на этом пути возникает задача декодирования кода $C_1 \otimes C_2$, решение которой в общем случае не известно даже тогда, когда известны эффективные декодеры для кодов C_1, C_2.

Целью настоящей работы является построение и обоснование набора необходимых для декодирования кода $C_1 \otimes C_2$ быстрых алгоритмов, когда C_1 и C_2 — групповые MLD-коды на группах $\mathcal{G}$ и $\mathcal{H}$ соответственно. Отметим, что в этом случае код $C_1 \otimes C_2$ является групповым кодом на прямом произведении $\mathcal{G} \times \mathcal{H}$. Процесс построения декодера существенно опирается на групповые свойства кода $C_1 \otimes C_2$ и на результаты работы [13], в которой построены алгоритмы декодирования для индуцированных групповых кодов. В качестве приложения построена криптосистема типа Мак-Элиса на коде $C_1 \otimes C_2$ и приводится теоретическая оценка ее стойкости к атаке на ключ в предположении, что для кодовых криптосистем на кодах C_i возможна эффективная атака на ключ. В работе полученные результаты о стойкости численно проиллюстрированы в случае, когда C_i — код Рида–Маллера–Бермана, определенный на аддитивной группе поля $\mathbb{F}_{2^{m_i}}$, $1 \leq m_i \leq 8$, для которого криптосистема типа Мак-Элиса взломана Л. Миндером и А. Шокроллахи (2007 г.).

Результаты работы представлены в первом и втором разделах. В первом разделе приводятся необходимые сведения о групповых MLD-кодах и далее строятся и обосновываются конструктивные алгоритмы для декодирования тензорного произ-

ведения таких кодов. Во втором разделе проводится анализ стойкости криптосистемы типа Мак-Элиса на коде $C_1 \otimes C_2$ к нахождению подходящего секретного ключа, если известна аналогичная структурная атака хотя бы для одного из кодов C_1 и C_2.

1. Тензорное произведение MLD-кодов

1.1. MLD-коды

Для натурального n символом $\underline{n}$ будем обозначать множество $\{1, ..., n\}$. Пусть V – векторное пространство над конечным полем $\mathbb{F}$. Зафиксируем в V базис B и символом (V, d_B) обозначим метрическое пространство V с метрикой Хэмминга d_B, построенной относительно базиса B. Для вектора $\mathbf{x}(\in V)$ множество базисных векторов, коэффициенты при которых в разложении $\mathbf{x} = \sum_{\mathbf{b}\in B} x_{\mathbf{b}}\mathbf{b}$ ненулевые, называется носителем вектора $\mathbf{x}$ относительно базиса B и обозначается $\mathrm{supp}_B(\mathbf{x})$; коэффициенты $x_{\mathbf{b}}$ будем называть значением $\mathbf{b}$-координаты вектора $\mathbf{x}$. Вес $w_B(\mathbf{x})$ вектора $\mathbf{x}$ определяется как $|\mathrm{supp}_B(\mathbf{x})|$. (Здесь и далее символом $|A|$ обозначается мощность множества A.) Всякое линейное подпространство C метрического пространства (V, d_B) называется линейным кодом. Размерность и длину кода будем обозначать соответственно $k(C)$ и $n(C)$, а минимальное кодовое расстояние кода C обозначим $\mathrm{dist}_B(C)$. Двойственный код к коду C обозначим $C^\perp$. Множество векторов $\mathcal{M}_{\mathbf{v}} = \{\mathbf{v}^{(1)}, ..., \mathbf{v}^{(r)}\}(\subset V)$ называется M-ортогональным вектору $\mathbf{v}(\in V)$, если $|\mathrm{supp}_B(\mathbf{v}^{(i)})| > |\mathrm{supp}_B(\mathbf{v})|$ для всех $i = 1, ..., r$ и

$$\forall\ i \neq j: \quad \mathrm{supp}_B(\mathbf{v}^{(i)}) \cap \mathrm{supp}_B(\mathbf{v}^{(j)}) = \mathrm{supp}_B(\mathbf{v}). \tag{1}$$

Пусть $\mathbf{c}(\in C)$ — кодовый вектор, $\mathbf{x} = \mathbf{c} + \mathbf{e}$ — принятый из канала вектор, $w_B(\mathbf{e}) \leq \lfloor(\mathrm{dist}_B(C) - 1)/2\rfloor$. Рассмотрим разложение $\sum_{\mathbf{b}\in B} e_{\mathbf{b}}\mathbf{b}$ вектора ошибок $\mathbf{e}$ по базису B. Если для $\mathbf{b}$-координаты существует *декодирующее дерево* $\mathrm{WB}_{\mathbf{b},r_{\mathbf{b}},L_{\mathbf{b}}}$, такое, что $\lfloor r_{\mathbf{b}}/2\rfloor \geq w_B(\mathbf{e})$, то значение $e_{\mathbf{b}}$ для $\mathbf{b}$-координаты вектора $\mathbf{e}$ находится однозначно с помощью мажоритарного декодера (см. [13], алгоритм 3 Decoder2). Декодирующим деревом $\mathrm{WB}_{\mathbf{b},r_{\mathbf{b}},L_{\mathbf{b}}} = \mathrm{WB}_{\mathbf{b},r_{\mathbf{b}},L_{\mathbf{b}}}[C]$ для $\mathbf{b}$-координаты здесь и далее будем называть в соответствии с [13] помеченное дерево с корнем $\mathbf{b}$, обладающее следующими свойствами:

1) множество вершин этого дерева состоит из $L_{\mathbf{b}} + 1$ уровня; корень с меткой $\mathbf{b}(\in B)$ находится на уровне 0, а листья – на уровне $L_{\mathbf{b}}$; метки вершин i-го уровня образуют набор V_i, $i = 0, ..., L_{\mathbf{b}}$;

2) листья дерева помечены элементами из $C^\perp$;

3) каждая вершина, не являющаяся листом, имеет не менее $r_{\mathbf{b}}(\in \mathbb{N})$ непосредственно следующих за ней вершин;

4) с меткой $\mathbf{p}$ каждой вершины дерева связывается числовое значение $l(\mathbf{p})(\in \mathbb{F})$ метки, вычисляемое в зависимости от значения принятого из канала вектора $\mathbf{x}$: для каждого листа $\mathbf{p}$ дерева значение $l(\mathbf{p})$ равно скалярному произведению $(\mathbf{p}, \mathbf{x})$ векторов $\mathbf{p}$ и $\mathbf{x}$, а для вершин на уровне $i(0 \leq i \leq L_{\mathbf{b}} - 1)$ значение $l(\mathbf{p})$ вычисляется в соответствии с построенным в [13] алгоритмом MajorVote (для полноты изложения алгоритм MajorVote приведен ниже);

5) метки вершин, непосредственно следующих из произвольной вершины $\mathbf{p}$, находящейся на уровне $i(0 \le i < L_{\mathbf{b}})$, образуют в совокупности множество $\mathcal{M}_{\mathbf{p}}$, M-ортогональное $\mathbf{p}$; символом $l[\mathcal{M}_{\mathbf{p}}]$ обозначается набор $(l(\mathbf{q}))_{\mathbf{q}\in\mathcal{M}_{\mathbf{p}}}$.

Исходные параметры: $\mathcal{A}$ – последовательность чисел из $\mathbb{F}$
Результат: элемент $v(\in \mathbb{F})$, который в последовательности $\mathcal{A}$ встречается наибольшее число раз
для каждого $a \in \mathbb{F}$ **выполнять**
  вычислить величину $\mathrm{count}(a)$, равную числу появления элемента a в последовательности $\mathcal{A}$
конец цикла
если *найдется только один* $a'(\in \mathbb{F})$, *что* $\mathrm{count}(a') \ge \lceil |\mathcal{A}|/2 \rceil$ **тогда**
  $v := a'$
иначе
  $v := 0$
конец условия
возвратить v

Алгоритм 1: MajorVote

Если для кода C существует такой набор

$$\mathcal{WB}(C) = \{\mathrm{WB}_{\mathbf{b},r_{\mathbf{b}},L_{\mathbf{b}}}\}_{\mathbf{b}\in B}, \tag{2}$$

для которого $\mathrm{dmaj}_B(C) = \min_{\mathbf{b}(\in B)}\{r_{\mathbf{b}}\} = \mathrm{dist}_B(C) - 1$, то код C называют MLD-кодом (Majority Logic Decodable). Заметим, что $\mathrm{dmaj}_B(C) \le \mathrm{dist}_B(C) - 1$, иначе в противном случае получили бы, что код может гарантированно исправлять более $\lfloor(\mathrm{dist}_B(C) - 1)/2\rfloor$ ошибок.

Построение набора $\mathcal{WB}(C)$ представляется в общем случае сложной задачей. С одной стороны, сложной представляется задача построения дерева для фиксированной координаты, а с другой стороны, деревья для разных координат строятся независимо. В то же время вторая задача решается просто для групповых кодов, если имеется декодирующее дерево хотя бы для одной координаты. Необходимые сведения о групповых кодах и построении множества $\mathcal{WB}(C)$ для группового кода C приводятся ниже.

1.2. Групповые MLD-коды

Пусть $\mathcal{G} = \{g_1 = \hat{1}, ..., g_{|\mathcal{G}|}\}$ — конечная группа с зафиксированным линейным порядком на множестве ее элементов, $\hat{1}$ – нейтральный элемент группы; зафиксированный порядок на группе будем обозначать $\mathrm{ord}(\mathcal{G})$. Рассмотрим групповую алгебру $\mathbb{F}\mathcal{G}$, элементами которой являются формальные суммы (функции):

$$\sum_{g\in\mathcal{G}} a_g g, a_g \in \mathbb{F}. \tag{3}$$

В конечномерной групповой алгебре $\mathbb{F}\mathcal{G}$ зафиксируем базис $B = B^{\mathbb{F}\mathcal{G}} := \{\mathbf{g} = \delta_g\}_{g\in\mathcal{G}}$, где $\delta_g = 1g$ – функция Дирака; $\mathbf{1} := 1\hat{1}$. Это позволяет рассматривать в $\mathbb{F}\mathcal{G}$ метрику Хэмминга d_B. Отметим, что в категории конечномерных линейных пространств

$\mathbb{F}\mathcal{G}$ и $\mathbb{F}^{|\mathcal{G}|}$ изоморфны; соответствующий изоморфизм обозначим $\nu_{\mathcal{G}}$. Также отметим, что произведение функций δ_x и δ_y в $\mathbb{F}\mathcal{G}$ равно $\delta_x\delta_y = \delta_{xy}$. Поэтому элементы групповой алгебры можно записывать в виде: $\sum_{g\in\mathcal{G}} a_g\delta_g, a_g \in \mathbb{F}$. Для удобства значение функции $\phi(\in \mathbb{F}\mathcal{G})$ в точке $g(\in \mathcal{G})$ будем обозначать $\phi(g)$.

В соответствии с [14], с. 39, всякий отличный от $\{\mathbf{0}\}$ *левый* идеал C в групповой алгебре $\mathbb{F}\mathcal{G}$ называется *групповым кодом* ($\mathbb{F}\mathcal{G}$–кодом) длины $n(C) = |\mathcal{G}|$. Идеал в групповой алгебре $\mathbb{F}\mathcal{G}$ является подпространством пространства функций $\mathbb{F}\mathcal{G}$, размерность $k(C)$ кода C — это размерность этого подпространства. Пусть $B^C = \{\epsilon_1, ..., \epsilon_{k(C)}\}(\subseteq \mathbb{F}\mathcal{G})$ — базис идеала C. Заметим, что порядок $\mathrm{ord}(\mathcal{G})$ индуцирует порядок на базисе $B^{\mathbb{F}\mathcal{G}}$ групповой алгебры $\mathbb{F}\mathcal{G}$, что позволяет выписать порождающую матрицу $G(C)$ группового кода C:

$$G(C) = \begin{pmatrix} \nu_{\mathcal{G}}(\epsilon_1) \\ \dots \\ \nu_{\mathcal{G}}(\epsilon_{k(C)}) \end{pmatrix} \tag{4}$$

Группа $\mathcal{G}$ действует слева на групповой алгебре $\mathbb{F}\mathcal{G}$ следующим естественным образом (см. [14], с. 32):

$$\mathcal{G} \times \mathbb{F}\mathcal{G} \ni (g, \phi = \sum_{h\in\mathcal{G}} \phi_h h) \mapsto \phi g^{-1} := \sum_{h\in\mathcal{G}} \phi_{hg^{-1}} h \in \mathbb{F}\mathcal{G}. \tag{5}$$

Отметим, что $C^\perp$ — также групповой код [14], т.е. левый идеал. В силу этого действие группы $\mathcal{G}$ по правилу (5) на элементах кода $C^\perp$ не выводит за код $C^\perp$. С другой стороны, группа $\mathcal{G}$ действует транзитивно на элементах из $B^{\mathbb{F}\mathcal{G}}$ и не нарушает M-ортогональности [13]. Это позволяет построить набор (2) по одному из декодирующих деревьев (соответствующие алгоритмы построены в [13]). В частности, если для базисной функции $\mathbf{1} = \delta_{\hat{1}} = 1\hat{1}$ удалось построить декодирующее дерево $\mathrm{WB}_{\mathbf{1},r_\mathbf{1},L_\mathbf{1}}$, то дерево $\mathrm{WB}_{\mathbf{g},r_\mathbf{g},L_\mathbf{g}}$ для базисной функции $\mathbf{g} = \delta_g = 1g$ может быть построено путем действия элементов $g^{-1}(\in \mathcal{G})$ на узлы дерева $\mathrm{WB}_{\mathbf{1},r_\mathbf{1},L_\mathbf{1}}$ по правилу (5). При этом $r_\mathbf{1} = r_\mathbf{g}$ и $L_\mathbf{1} = L_\mathbf{g}$ для всех $g \in \mathcal{G}$.

1.3. Тензорное произведение групповых кодов

Пусть $\mathcal{G} = \{g_1, ..., g_{|\mathcal{G}|}\}$, $\mathcal{H} = \{h_1, ..., h_{|\mathcal{H}|}\}$ — конечные группы с зафиксированными на них линейными порядками $\mathrm{ord}(\mathcal{G})$ и $\mathrm{ord}(\mathcal{H})$. В групповых алгебрах $\mathbb{F}\mathcal{G}$ и $\mathbb{F}\mathcal{H}$ зафиксируем базисы $B^{\mathbb{F}\mathcal{G}} = \{\delta_{g_1}, ..., \delta_{g_{|\mathcal{G}|}}\}$ и $B^{\mathbb{F}\mathcal{H}} = \{\delta_{h_1}, ..., \delta_{h_{|\mathcal{H}|}}\}$ соответственно. Рассмотрим тензорное произведение $\mathbb{F}\mathcal{G}\otimes_{\mathbb{F}}\mathbb{F}\mathcal{H}$ групповых алгебр $\mathbb{F}\mathcal{G}$ и $\mathbb{F}\mathcal{H}$ над полем $\mathbb{F}$ (см. [15], с.79). Отметим, что $\mathbb{F}\mathcal{G} \otimes_{\mathbb{F}} \mathbb{F}\mathcal{H} = \mathbb{F}(\mathcal{G} \times \mathcal{H})$.

В групповых алгебрах $\mathbb{F}\mathcal{G}$ и $\mathbb{F}\mathcal{H}$ рассмотрим групповые коды C_1 ($\subseteq \mathbb{F}\mathcal{G}$) и C_2 ($\subseteq \mathbb{F}\mathcal{H}$) с соответствующими базисами $B^{C_1} = \{\epsilon_1, ..., \epsilon_{k(C_1)}\}$ и $B^{C_2} = \{\phi_1, ..., \phi_{k(C_2)}\}$, где $\epsilon_i = \sum_{g\in\mathcal{G}} a_{i,g}\delta_g, a_{i,g} \in \mathbb{F}$ и $\phi_j = \sum_{h\in\mathcal{H}} b_{j,h}\delta_h, b_{j,h} \in \mathbb{F}$. Тензорным произведением кодов C_1 и C_2 будем называть код $C_1 \otimes C_2(\subseteq \mathbb{F}(\mathcal{G} \times \mathcal{H}))$ с базисом $B^{C_1\otimes C_2} = \{\epsilon_i \otimes \phi_j | i = 1, ..., k(C_1), j = 1, ..., k(C_2)\}$, где $(\epsilon_i \otimes \phi_j)(g, h) = \epsilon_i(g)\phi_j(h)$, $(g, h) \in \mathcal{G} \times \mathcal{H}$. Под тензорным произведением $A \otimes B$ матрицы $A = (a_{i,j})$ размера $(r \times s)$ и матрицы B

будем понимать, как обычно, матрицу вида:

$$A \otimes B = \begin{pmatrix} a_{1,1}B & \dots & a_{1,s}B \\ a_{2,1}B & \dots & a_{2,s}B \\ \dots & \dots & \dots \\ a_{r,1}B & \dots & a_{r,s}B \end{pmatrix}.$$

Тогда $G(C_1 \otimes C_2) = G(C_1) \otimes G(C_2)$ — порождающая матрица кода $C_1 \otimes C_2$, где, согласно (4), порождающие матрицы кодов C_1 и C_2 могут быть представлены в виде соответственно:

$$G(C_1) = \begin{pmatrix} a_{1,g_1} & \dots & a_{1,g_{|\mathcal{G}|}} \\ a_{2,g_1} & \dots & a_{2,g_{|\mathcal{G}|}} \\ \dots & \dots & \dots \\ a_{k(C_1),g_1} & \dots & a_{k(C_1),g_{|\mathcal{G}|}} \end{pmatrix}, \quad G(C_2) = \begin{pmatrix} b_{1,h_1} & \dots & b_{1,h_{|\mathcal{H}|}} \\ b_{2,h_1} & \dots & b_{2,h_{|\mathcal{H}|}} \\ \dots & \dots & \dots \\ b_{k(C_2),h_1} & \dots & b_{k(C_2),h_{|\mathcal{H}|}} \end{pmatrix}.$$

Заметим, $k(C_1 \otimes C_2) = k(C_1)k(C_2)$, $n(C_1 \otimes C_2) = n(C_1)n(C_2)$ и

$$\operatorname{dist}_{B^{\mathbb{F}(\mathcal{G}\times\mathcal{H})}}(C_1 \otimes C_2) = \operatorname{dist}_{B^{\mathbb{F}\mathcal{G}}}(C_1)\operatorname{dist}_{B^{\mathbb{F}\mathcal{H}}}(C_2), \tag{6}$$

где $B^{\mathbb{F}(\mathcal{G}\times\mathcal{H})} = \{\delta_{(g_1,h_1)}, \dots, \delta_{(g_1,h_{|\mathcal{H}|})}, \delta_{(g_2,h_1)}, \dots, \delta_{(g_{|\mathcal{G}|},h_{|\mathcal{H}|})}\}$ — базис групповой алгебры $\mathbb{F}(\mathcal{G} \times \mathcal{H})$ на группе $\mathcal{G} \times \mathcal{H}$ с линейным порядком, индуцированным линейными порядками $\operatorname{ord}(\mathcal{G})$ и $\operatorname{ord}(\mathcal{H})$.

1.4. Декодирование тензорного произведения MLD-кодов

Пусть $\mathbf{c}(\in C_1 \otimes C_2)$ – кодовый вектор, который на выходе из канала принимает вид

$$\mathbf{x} = \mathbf{c} + \mathbf{e}, \quad \mathbf{e} \in \mathbb{F}(\mathcal{G} \times \mathcal{H}). \tag{7}$$

В этом разделе строятся алгоритмы, позволяющие правильно находить значение вектора ошибок $\mathbf{e}$, если вес этого вектора удовлетворяет следующему неравенству (см. (6)):

$$w_{B^{\mathbb{F}(\mathcal{G}\times\mathcal{H})}}(\mathbf{e}) \le \left\lfloor \frac{\operatorname{dist}_{B^{\mathbb{F}(\mathcal{G}\times\mathcal{H})}}(C_1 \otimes C_2) - 1}{2} \right\rfloor. \tag{8}$$

Прежде всего сформулируем вспомогательную лемму.

Лемма 1. *Пусть* $\mathbf{c}_1 \in \mathbb{F}^{n_1}$, $\mathcal{M}_{\mathbf{c}_1}$ – *M-ортогональное множество для вектора* $\mathbf{c}_1$, $\mathbf{c}_2 \in \mathbb{F}^{n_2}$, $\mathcal{M}_{\mathbf{c}_2}$ – *M-ортогональное множество для вектора* $\mathbf{c}_2$, *тогда M-ортогональное множество для вектора* $\mathbf{c}_1 \otimes \mathbf{c}_2$ *состоит из векторов вида:*

$$1)\mathbf{c}_1 \otimes \mathbf{w},$$
$$2)\mathbf{v} \otimes \mathbf{c}_2,$$
$$3)\mathbf{c}_1 \otimes \mathbf{w} + \mathbf{v} \otimes \mathbf{c}_2 + \mathbf{v} \otimes \mathbf{w},$$

где $\mathbf{v} \in \mathcal{M}_{\mathbf{c}_1}$, $\mathbf{w} \in \mathcal{M}_{\mathbf{c}_2}$.

Доказательство. Эта лемма вытекает из [14], с. 121–122. □

Для векторов $\mathbf{c}_1 \in C_1(\subseteq \mathbb{F}\mathcal{G})$ и $\mathbf{c}_2 \in C_2(\in\subseteq \mathbb{F}\mathcal{H})$ рассмотрим соответствующие им M-ортогональные множества $\mathcal{M}_{\mathbf{c}_1}$ и $\mathcal{M}_{\mathbf{c}_2}$. Ниже построен алгоритм M_orth, который конструирует для каждого вектора $\mathbf{c}_1 \otimes \mathbf{c}_2 \in C_1 \otimes C_2(\subseteq \mathbb{F}(\mathcal{G} \times \mathcal{H}))$ такое M-ортогональное множество $\mathcal{M}_{\mathbf{c}_1\otimes\mathbf{c}_2}$, что

$$|\mathcal{M}_{\mathbf{c}_1\otimes\mathbf{c}_2}| = |\mathcal{M}_{\mathbf{c}_1}| + |\mathcal{M}_{\mathbf{c}_1}|. \tag{9}$$

Исходные параметры: Векторы $\mathbf{c}_1$,$\mathbf{c}_2$ и соответствующие им
$\mathcal{M}$-ортогональные множества $\mathcal{M}_{\mathbf{c}_1}$, $\mathcal{M}_{\mathbf{c}_2}$.
Результат: $\mathcal{M}$-ортогональное множество $\mathcal{M}_{\mathbf{c}_1\otimes\mathbf{c}_2}$ для вектора $\mathbf{c}_1 \otimes \mathbf{c}_2$.
$\mathcal{M}_{\mathbf{c}_1\otimes\mathbf{c}_2} := \emptyset$
для каждого $\mathbf{w} \in \mathcal{M}_{\mathbf{c}_2}$ **выполнять**
| к $\mathcal{M}_{\mathbf{c}_1\otimes\mathbf{c}_2}$ добавить $\mathbf{c}_1 \otimes \mathbf{w}$
конец цикла
для каждого $\mathbf{v} \in \mathcal{M}_{\mathbf{c}_1}$ **выполнять**
| к $\mathcal{M}_{\mathbf{c}_1\otimes\mathbf{c}_2}$ добавить $\mathbf{v} \otimes \mathbf{c}_2$
конец цикла
возвратить $\mathcal{M}_{\mathbf{c}_1\otimes\mathbf{c}_2}$

Алгоритм 2: M_orth

С применением алгоритма M_orth построен алгоритм MakeTensorTree, который для корня с меткой $\mathbf{1}_\mathcal{G} \otimes \mathbf{1}_\mathcal{H}$ по декодирующим деревьям $\mathrm{WB}_{\mathbf{1}_\mathcal{G},r_{\mathbf{1}_\mathcal{G}},L_{\mathbf{1}_\mathcal{G}}}[C_1]$ и $\mathrm{WB}_{\mathbf{1}_\mathcal{H},r_{\mathbf{1}_\mathcal{H}},L_{\mathbf{1}_\mathcal{H}}}[C_2]$ строит некоторое *вспомогательное декодирующее дерево*, которое обозначим следующим образом:

$$\mathrm{WB}^{\otimes}{}_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H},r_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}},L_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}}}[C_1 \otimes C_2]. \tag{10}$$

Поясним, почему дерево, строящееся в алгоритме MakeTensorTree, имеет $L_{\mathbf{1}_\mathcal{G}} + L_{\mathbf{1}_\mathcal{H}} - 1$ уровней. В этом алгоритме с помощью алгоритма M_orth для каждой вершины $\mathbf{v} = \mathbf{c}_1^i \otimes \mathbf{c}_2^j$ на уровне k дерева (10), где $i \in \{0, ..., L_{\mathbf{1}_\mathcal{G}} - 1\}$ – i-й уровень декодирующего дерева для кода C_1, $j \in \{0, ..., L_{\mathbf{1}_\mathcal{H}} - 1\}$ – j-й уровень декодирующего дерева для кода C_2, строится множество векторов на уровне $k + 1$ дерева (10), состоящее из подмножеств двух типов:

1) $\mathbf{c}_1^i \otimes \mathbf{c}_2^{j+1}$,
2) $\mathbf{c}_1^{i+1} \otimes \mathbf{c}_2^j$.

Таким образом, максимальный уровень дерева (10) будет достигаться, например, если сначала поочередно пройти $L_{\mathbf{1}_\mathcal{G}} - 1$ векторов второго типа, каждый из которых находится на следующем уровне дерева, затем пройти $L_{\mathbf{1}_\mathcal{H}}$ векторов первого типа. Тогда глубина дерева (10) будет равна $L_{\mathbf{1}_\mathcal{G}} + L_{\mathbf{1}_\mathcal{H}} - 1$.

Если C_1 и C_2 — групповые MLD-коды, то есть $\mathrm{dmaj}_{B^{\mathbb{F}\mathcal{G}}}(C_1) = \mathrm{dist}_{B^{\mathbb{F}\mathcal{G}}}(C_1) - 1$ и $\mathrm{dmaj}_{B^{\mathbb{F}\mathcal{G}}}(C_2) = \mathrm{dist}_{B^{\mathbb{F}\mathcal{G}}}(C_2) - 1$, то в общем случае вспомогательное декодирующее дерево, построенное по алгоритму MakeTensorTree, не позволяет найти с помощью мажоритарного декодера (см. [13], алгоритм 3 Decoder2) значения ошибок $\mathbf{e}$, вес которых удовлетворяет неравенству (8). Дело в том, что из сравнения равенств (6) и

Исходные параметры: $\mathrm{WB}_{\mathbf{1}_\mathcal{G},r_{\mathbf{1}_\mathcal{G}},L_{\mathbf{1}_\mathcal{G}}}[C_1]$, $\mathrm{WB}_{\mathbf{1}_\mathcal{H},r_{\mathbf{1}_\mathcal{H}},L_{\mathbf{1}_\mathcal{H}}}[C_2]$.
Результат: дерево (10).
$V_1^{\otimes} := \mathrm{M\_orth}(\mathbf{1}_\mathcal{G}, \mathbf{1}_\mathcal{H}, \mathcal{M}_{\mathbf{1}_\mathcal{G}}, \mathcal{M}_{\mathbf{1}_\mathcal{H}})$
цикл $1 \le k \le L_{\mathbf{1}_\mathcal{G}} + L_{\mathbf{1}_\mathcal{H}} - 2$ **выполнять**
  для каждого $\mathbf{v}^k = (\mathbf{c}_1^k \otimes \mathbf{c}_2^k) \in V_k^{\otimes}$ **выполнять**
    если $\mathbf{v}^k \notin (C_1 \otimes C_2)^{\perp}$ **тогда**
      на уровень $V_{k+1}^{\otimes}$ добавить $|\mathcal{M}_{\mathbf{c}_1^k}| \cdot |\mathcal{M}_{\mathbf{c}_2^k}|$ вершин и соединить их с вершиной, имеющей метку $\mathbf{v}^k$ на уровне $V_k^{\otimes}$;
      пометить добавленные вершины метками из множества

$$\mathcal{M}_{\mathbf{v}^k} = \mathrm{M\_orth}(\mathbf{c}_1^k, \mathbf{c}_2^k, \mathcal{M}_{\mathbf{c}_1^k}, \mathcal{M}_{\mathbf{c}_2^k}).$$

    конец условия
  конец цикла
конец цикла
возвратить $\mathrm{WB}^{\otimes}_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H},r_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}},L_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}}}[C_1 \otimes C_2]$

Алгоритм 3: MakeTensorTree

(9) вытекает, что мощность $|\mathcal{M}_{\mathbf{c}_1}|+|\mathcal{M}_{\mathbf{c}_2}|$ построенного алгоритмом MakeTensorTree M-ортогонального множества $\mathcal{M}_{\mathbf{c}_1\otimes\mathbf{c}_2}$ меньше

$$\mathrm{dist}_{B^{\mathbb{F}(\mathcal{G}\times\mathcal{H})}}(C_1 \otimes C_2) - 1 = |\mathcal{M}_{\mathbf{c}_1}||\mathcal{M}_{\mathbf{c}_2}| + |\mathcal{M}_{\mathbf{c}_1}| + |\mathcal{M}_{\mathbf{c}_2}|$$

для каждого узла с меткой $\mathbf{c}_1 \otimes \mathbf{c}_2$.

Вспомогательное декодирующее дерево (10) может быть достроено до полного декодирующего дерева MLD-кода $C_1 \otimes C_2$ на основании конструкции 3) леммы 1 путем добавления недостающих вершин, однако представляется удобным работать только со значениями меток недостающих вершин. Алгоритм AddVals для каждого узла $\mathbf{v} = \mathbf{c}_1 \otimes \mathbf{c}_2$ по принятому из канала вектору $\mathbf{x}$ и вспомогательному декодирующему дереву (10) вычисляет дополнительные $|\mathcal{M}_{\mathbf{c}_1}||\mathcal{M}_{\mathbf{c}_2}|$ значений меток недостающих вершин, но, подчеркнем, к дереву (10) при выполнении алгоритма AddVals дополнительные вершины не добавляются.

Алгоритм AddVals применяется в алгоритме декодирования DecodeTensorBit, который по принятому вектору $\mathbf{x}$ находит значение $e_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}}$ вектора ошибок $\mathbf{e}$ в координате, соответствующей базисной функции $\mathbf{1}_\mathcal{G} \otimes \mathbf{1}_\mathcal{H}$. (Отметим, что в алгоритме DecodeTensorBit используется операция конкатенации наборов чисел, которая обозначается символом $\uplus$.) Таким образом, алгоритм DecodeTensorBit в случае тензорного произведения кодов выполняет функцию упомянутого выше алгоритма мажоритарного декодирования и правильно находит значение координаты вектора ошибок $\mathbf{e}$, когда вес ошибки удовлетворяет неравенству (8).

Заметим, что алгоритм MakeTensorTree строит вспомогательное декодирующее дерево для координаты, соответствующей элементу $\widehat{1}_\mathcal{G}\times\widehat{1}_\mathcal{H}(\in \mathcal{G}\times\mathcal{H})$. В [13] для групповых кодов построен приведенный ниже вспомогательный алгоритм CloneTree, позволяющий по вспомогательному декодирующему дереву с одной меткой у корня построить вспомогательноее декодирующее дерево для корня с любой другой меткой.

Исходные параметры: $\mathbf{x}$ – вектор вида (7), $\mathcal{M}_{\mathbf{v}^k}$ – M - ортогональное множество для $\mathbf{v}^k = \mathbf{c}_1^k \otimes \mathbf{c}_2^k$, $\mathrm{WB}^{\otimes}{}_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}, r_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}}, L_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}}}[C_1 \otimes C_2]$ – декодирующее дерево, k – текущий уровень дерева
Результат: Набор чисел l_k мощности $|\mathcal{M}_{\mathbf{c}_1^k}| \cdot |\mathcal{M}_{\mathbf{c}_2^k}|$
если $k = L_{\mathbf{1}_\mathcal{G}} + L_{\mathbf{1}_\mathcal{H}} - 2$ **тогда**
  цикл $1 \le i \le |\mathcal{M}_{\mathbf{c}_2^k}|$ **выполнять**
    цикл $|\mathcal{M}_{\mathbf{c}_2^k}| + 1 \le j \le |\mathcal{M}_{\mathbf{c}_2^k}| + |\mathcal{M}_{\mathbf{c}_1^k}|$ **выполнять**
      к l_k добавить $l(\mathbf{v}_i = (\mathbf{c}_1^i \otimes \mathbf{c}_2^i)) + l(\mathbf{v}_j = (\mathbf{c}_1^j \otimes \mathbf{c}_2^j)) + < \mathbf{c}_1^j \otimes \mathbf{c}_2^i, \mathbf{x} >$, где $\mathbf{v}_i, \mathbf{v}_j \in \mathcal{M}_{\mathbf{v}^k}, \mathbf{c}_1^j \otimes \mathbf{c}_2^i \in (C_1 \otimes C_2)^\perp$
    конец цикла
  конец цикла
иначе
  цикл $1 \le i \le |\mathcal{M}_{\mathbf{c}_2^k}|$ **выполнять**
    цикл $|\mathcal{M}_{\mathbf{c}_2^k}| + 1 \le j \le |\mathcal{M}_{\mathbf{c}_2^k}| + |\mathcal{M}_{\mathbf{c}_1^k}|$ **выполнять**
      к l_k добавить $l(\mathbf{v}_i = (\mathbf{c}_1^i \otimes \mathbf{c}_2^i)) + l(\mathbf{v}_j = (\mathbf{c}_1^j \otimes \mathbf{c}_2^j)) + l(\mathbf{c}_1^j \otimes \mathbf{c}_2^i)$, где $\mathbf{v}_i, \mathbf{v}_j \in \mathcal{M}_{\mathbf{v}^k}, \mathbf{c}_1^j \otimes \mathbf{c}_2^i \in V^{\otimes}{}_{k+2}$
    конец цикла
  конец цикла
конец условия
возвратить l_k

Алгоритм 4: AddVals

Исходные параметры: $\mathbf{x}$ – вектор вида (7), $\mathrm{WB}^{\otimes}[C_1 \otimes C_2] = \mathrm{WB}^{\otimes}{}_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}, r_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}}, L_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}}}[C_1 \otimes C_2]$ – декодирующее дерево
Результат: $e_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}}$
цикл $1 \le k \le L_{\mathbf{1}_\mathcal{G}} + L_{\mathbf{1}_\mathcal{H}} - 1$ **выполнять**
  для каждого $\mathbf{v}_k \in V^{\otimes}{}_k$ **выполнять**
    если $\mathbf{v}_k \in (C_1 \otimes C_2)^\perp$ **тогда**
      $l(\mathbf{v}_k) :=< \mathbf{v}_k, \mathbf{x} >$
    иначе
      $l(\mathbf{v}_k) := \mathrm{MajorVote}(l[\mathcal{M}_{v^k}] \uplus \mathrm{AddVals}(\mathbf{x}, \mathcal{M}_{v^k}, \mathrm{WB}^{\otimes}[C_1 \otimes C_2], k))$
    конец условия
  конец цикла
конец цикла
$e_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}} := \mathrm{MajorVote}(l[\mathcal{M}_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}}] \uplus \mathrm{AddVals}(\mathbf{x}, \mathcal{M}_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}}, \mathrm{WB}^{\otimes}[C_1 \otimes C_2], 0))$
возвратить $e_{\mathbf{1}_\mathcal{G}\otimes\mathbf{1}_\mathcal{H}}$

Алгоритм 5: DecodeTensorBit

Таким образом, набор вспомогательных декодирующих деревьев

$$\mathcal{WB}^{\otimes}(C_1 \otimes C_2) = \{\mathrm{WB}^{\otimes}_{\delta_{(g,h)}, r_{\delta_{(g,h)}}, L_{\delta_{(g,h)}}}\}_{\delta_{(g,h)} \in B^{\mathbb{F}(\mathcal{G}\times\mathcal{H})}}$$

для группового кода $C_1 \otimes C_2$ может быть построен по дереву (10). Именно, для

Исходные параметры: $\mathcal{G}$, $\mathrm{WB}^{\otimes}_{\mathbf{b},r_{\mathbf{b}},L_{\mathbf{b}}}[C]$, $\mathbf{b}'$
Результат: $\mathrm{WB}^{\otimes}_{\mathbf{b}',r'_{\mathbf{b}},L_{\mathbf{b}'}}[C]$
$\mathrm{WB}^{\otimes}_{\mathbf{b}',r'_{\mathbf{b}},L_{\mathbf{b}'}}[C] := \mathrm{WB}^{\otimes}_{\mathbf{b},r_{\mathbf{b}},L_{\mathbf{b}}}[C]$;
Найти $g(\in \mathcal{G})$ такой, что $(g,\mathbf{b}) = \mathbf{b}'$;
для каждой *метки* $\mathbf{p}$ *дерева* $\mathrm{WB}^{\otimes}_{\mathbf{b}',r'_{\mathbf{b}},L_{\mathbf{b}'}}[C]$ **выполнять**
  // Действие элементом g на $\mathbf{p}$ по правилу (5);
  $\mathbf{p} := (g,\mathbf{p})$;
конец цикла
возвратить $\mathrm{WB}^{\otimes}_{\mathbf{b}',r'_{\mathbf{b}},L_{\mathbf{b}'}}[C]$

Алгоритм 6: CloneTree

любой базисной функции $\delta_{(g,h)} \in B^{\mathbb{F}(\mathcal{G}\times\mathcal{H})}$:

$$\mathrm{WB}^{\otimes}_{\delta_{(g,h)},r_{\delta_{(g,h)}},L_{\delta_{(g,h)}}} = \mathrm{CloneTree}(\mathcal{G}\times\mathcal{H}, \mathrm{WB}^{\otimes}{}_{\mathbf{1}_{\mathcal{G}}\otimes\mathbf{1}_{\mathcal{H}},r_{\mathbf{1}_{\mathcal{G}}\otimes\mathbf{1}_{\mathcal{H}}},L_{\mathbf{1}_{\mathcal{G}}\otimes\mathbf{1}_{\mathcal{H}}}}, \delta_{(g,h)}).$$

По аналогии с алгоритмом Decoder3 из [13] построен алгоритм DecodeTensorVector декодирования принятого вектора $\mathbf{x}$, в котором каждая координата декодируется с помощью алгоритма DecodeTensorBit.

Исходные параметры: принятый вектор $\mathbf{x} = \mathbf{c} + \mathbf{e}$, набор вспомогательных декодирующих деревьев
$\mathcal{WB}^{\otimes}(C_1 \otimes C_2) = \{\mathrm{WB}^{\otimes}_{\mathbf{b},r_{\mathbf{b}},L_{\mathbf{b}}}\}_{\mathbf{b}\in B^{\mathbb{F}(\mathcal{G}\times\mathcal{H})}}$
Результат: вектор $\mathbf{c}'$ – результат декодирования
для каждого $\mathbf{b} \in B^{\mathbb{F}(\mathcal{G}\times\mathcal{H})}$ **выполнять**
  $a := x_{\mathbf{b}} - \mathrm{DecodeTensorBit}(\mathbf{x}, \mathbf{b}_i, \mathrm{WB}^{\otimes}_{\mathbf{b},r_{\mathbf{b}},L_{\mathbf{b}}})$
конец цикла
возвратить $\mathbf{c}'$

Алгоритм 7: DecodeTensorVector

Таким образом справедлива следующая теорема.

Теорема 1. *Пусть* C_1 *и* C_2 *— групповые* MLD*-коды с наборами декодирующих деревьев* $\mathcal{WB}(C_1)$ *и* $\mathcal{WB}(C_2)$ *соответственно,* $\mathcal{WB}^{\otimes}(C_1 \otimes C_2)$ *– набор вспомогательных декодирующих деревьев кода* $C_1 \otimes C_2$*, построенный с помощью алгоритмов* MakeTensorTree *и* CloneTree *с использованием* $\mathcal{WB}(C_1)$ *и* $\mathcal{WB}(C_2)$*. Тогда, если* $\mathbf{c}(\in C_1 \otimes C_2)$ *– кодовый вектор, который на выходе из канала принимает вид* $\mathbf{x} = \mathbf{c} + \mathbf{e}$*, где вес вектора ошибок* $\mathbf{e}$ *удовлетворяет условию* (8)*, то*

$$\mathrm{DecodeTensorVector}(\mathbf{x}, \mathcal{WB}^{\otimes}(C_1 \otimes C_2)) = \mathbf{c}.$$

2. Криптосистема типа Мак-Элиса на основе произведения кодов

2.1. Криптосистема типа Мак-Элиса

Пусть $C(\subseteq \mathbb{F}^n)$ – линейный $[n,k,d]$-код длины $n = n(C)$, размерности $k = k(C)$, с кодовым расстоянием $d = \mathrm{dist}_B(C)$, $G(C)$ – порождающая матрица кода C. Под крип-

тосистемой типа Мак-Элиса на основе $[n, k, d]$-кода C здесь понимается асимметричная криптосистема, в которой открытый ключ $\mathbf{k}_{\text{pub}}$ – это пара $(\widetilde{G}, t = \lfloor (d-1)/2 \rfloor)$, а секретный ключ $\mathbf{k}_{\text{sec}}$ – пара матриц (S, P), где S – случайная невырожденная $(k \times k)$-матрица, P – случайная перестановочная $(n \times n)$-матрица, причем $\widetilde{G} = S \cdot G(C) \cdot P$. Правило шифрования произвольного сообщения $\mathbf{s} (\in \mathbb{F}^k)$ имеет вид:

$$\mathbf{z} = \mathbf{s}\widetilde{G} + \mathbf{e}, \tag{11}$$

где вес Хэмминга добавляемой ошибки $\mathbf{e} = (e_1, ..., e_n)$ удовлетворяет неравенству: $w_B(\mathbf{e}) \leq t$. Для расшифрования $\mathbf{c}$ секретный ключ $\mathbf{k}_{\text{sec}}$ используется по правилу: $\mathbf{s} = \text{Dec}_C(\mathbf{z}P^{-1})S^{-1}$, где $\text{Dec}_C : \mathbb{F}^n \to \mathbb{F}^k$ — декодер кода C, гарантированно исправляющий t и менее ошибок и восстанавливающий вектор $\mathbf{s}$. Далее предполагается, что вектор ошибок $\mathbf{e}$ выбирается случайно и равновероятно из множества $\mathbb{F}^{n,t} = \mathbb{F}_q^{n,t} (\subseteq \mathbb{F}_q^n)$, состоящего из векторов веса t, $|\mathbb{F}^{n,t}| = C_n^t (q-1)^t$.

2.2. Анализ стойкости $\text{McE}(C_1 \otimes C_2)$ к атакам на ключ

Рассмотрим криптосистему типа Мак-Элиса $\text{McE}(C)$, где $C = C_1 \otimes C_2$ – тензорное произведение $[n_1, k_1, d_1]$-кода C_1 и $[n_2, k_2, d_2]$-кода C_2. В качестве модели нарушителя рассмотрим противника, целью которого является нахождение подходящего секретного ключа для правильного расшифрования криптограмм. Предполагается, что наблюдатель имеет алгоритм Attack, с помощью которого может быть эффективно найден подходящий секретный ключ для криптосистемы $\text{McE}(C_2)$.

Порождающая матрица кода $C = C_1 \otimes C_2$ имеет вид $G(C) = G(C_1) \otimes G(C_2)$, а размерность K кода C равна $k_1 k_2$. Тогда $(k_1 k_2 \times k_1 k_2)$-матрица S (часть секретного ключа $\mathbf{k}_{\text{sec}}$) может быть представлена в блочном виде:

$$S = \left(\begin{array}{c|c|c|c} S_{0,0} & S_{0,1} & ... & S_{0,k_1-1} \\ \hline S_{1,0} & S_{1,1} & ... & S_{1,k_1-1} \\ \hline ... & ... & ... & ... \\ \hline S_{k_1-1,0} & S_{k_1-1,1} & ... & S_{k_1-1,k_1-1} \end{array} \right), \tag{12}$$

где S_{ij} — $(k_2 \times k_2)$ -матрица, $i, j = 0, ..., k_1 - 1$. Поэтому для матрицы $S \cdot G(C)$ имеет место представление:

$$S \cdot G(C) = \left(\begin{array}{c|c|c} \sum\limits_{j=0}^{k_1-1} S_{0,j} g_{j,1}^1 G(C_2) & ... & \sum\limits_{j=0}^{k_1-1} S_{0,j} g_{j,n_1}^1 G(C_2) \\ \hline \sum\limits_{j=0}^{k_1-1} S_{1,j} g_{j,1}^1 G(C_2) & ... & \sum\limits_{j=0}^{k_1} S_{1,j} g_{j,n_1}^1 G(C_2) \\ \hline ... & ... & ... \\ \hline \sum\limits_{j=0}^{k_1-1} S_{k_1-1,j} g_{j,1}^1 G(C_2) & ... & \sum\limits_{j=0}^{k_1-1} S_{k_1-1,j} g_{jn_1}^1 G(C_2) \end{array} \right). \tag{13}$$

Для каждого $i \in \{1, ..., n_1\}$ блочный столбец матрицы (13) представим в виде:

$$\left(\begin{array}{c} \sum\limits_{j=0}^{k_1-1} S_{0,j} g_{j,i}^1 G(C_2) \\ \hline \sum\limits_{j=0}^{k_1-1} S_{1,j} g_{j,i}^1 G(C_2) \\ \hline \ldots \\ \hline \sum\limits_{j=0}^{k_1-1} S_{v_1,j} g_{j,i}^1 G(C_2) \end{array}\right) = \mathbf{S}_i G(C_2), \quad \mathbf{S}_i = \left(\begin{array}{c} S_{0,0} \\ \hline S_{1,0} \\ \hline \ldots \\ \hline S_{k_1-1,0} \end{array}\right) g_{0,i}^1 + \ldots + \left(\begin{array}{c} S_{0,k_1-1} \\ \hline S_{1,k_1-1} \\ \hline \ldots \\ \hline S_{k_1-1,k_1-1} \end{array}\right) g_{k_1-1,i}^1. \tag{14}$$

Непосредственно проверяется, что для каждого $i \in \{1, ..., n_1\}$ матрица $\mathbf{S}_i$ имеет ранг k_2. Тогда сложность нахождения подходящего секретного ключа не превышает соответствующей сложности применения алгоритма AttackInduced из [12] для криптосистемы типа Мак-Элиса на индуцированном коде с порождающей матрицей $I_{n_1} \otimes G_{k_2-1}^2$. Именно, если Q – вычислительная сложность алгоритма Attack для $\mathrm{McE}(C_2)$, то

$$\mathcal{O}\left(\left(\frac{n_1^{n_2-1}}{e}\right)^{n_1} (n_1 Q + k_1 k_2)^3\right) \tag{15}$$

– оценка сверху на сложность нахождения ключа для $\mathrm{McE}(C_1 \otimes C_2)$.

Таблица 1. Значения величины $\lceil \log_2(K(n_1, n_2)) \rceil$, где $K(n_1, n_2)$ – количество перебираемых ключей в атаке AttackInduced на криптосистему $\mathrm{McE}(C_1 \otimes C_2)$, здесь C_i – код Рида–Маллера $\mathcal{RM}(r_i, m_i)$, $n_i = 2^{m_i}$, $r_i \le m_i$, $m_i = 1, ..., 8$, $i \in \{1; 2\}$
Table 1. Values of $\lceil \log_2(K(n_1, n_2)) \rceil$, where $K(n_1, n_2)$ is number of probed keys in attack AttackInduced for $\mathrm{McE}(C_1 \otimes C_2)$ cryptosystem, where here C_i – Reed-Muller code $\mathcal{RM}(r_i, m_i)$, $n_i = 2^{m_i}$, $r_i \le m_i$, $m_i = 1, ..., 8$, $i \in \{1; 2\}$

	n_2							
n_1	*2*	*4*	*8*	*16*	*32*	*64*	*128*	*256*
2	0	0	8	25	58	122	**250**	**506**
4	0	16	48	112	**240**	**496**	**1008**	**2032**
8	8	48	**152**	**344**	**728**	**1496**	**3032**	**6104**
16	25	112	**344**	**928**	**1952**	**4000**	**8096**	**16288**
32	52	**240**	**728**	**1952**	**4896**	**10016**	**20256**	**40000**
64	122	**496**	**1496**	**4000**	**10016**	**24064**	**48640**	**96000**
128	**250**	**1008**	**3032**	**8096**	**20256**	**48640**	**113536**	**237000**
256	**506**	**2032**	**6104**	**16288**	**40000**	**96000**	**237000**	**512000**

В таблице 1 приведен пример расчета сложности атаки на ключ для криптосистемы $\mathrm{McE}(C_1 \otimes C_2)$, где C_i – код Рида–Маллера, $i \in \{1; 2\}$. В каждой ячейке таблицы приведено число $\lceil \log_2(K(n_1, n_2)) \rceil$, где $K(n_1, n_2)$ – количество перебираемых ключей в атаке AttackInduced (множитель $(n_1^{n_2-1} e^{-1})^{n_1}$ в (15)), где n_1 и n_2 – длины кодов C_1 и C_2 соответственно. Жирным выделены те ячейки таблицы, для которых $\lceil \log_2(K(n_1, n_2)) \rceil \ge 128$, так как перебор ключей длины 128 бит и более

в настоящее время является вычислительно неосуществимой задачей [16]. Заметим также, что $A \otimes B = Q_1 \cdot (B \otimes A) \cdot Q_2$, где Q_1 и Q_2 – перестановочные матрицы подходящего размера. Поэтому парам (n_1, n_2) и (n_2, n_1) в таблице соответствуют ячейки с одинаковым значением $\min\{\lceil \log_2(K(n_1, n_2)) \rceil ; \lceil \log_2(K(n_2, n_1)) \rceil\}$. На основании результатов, представленных в таблице 1, можно сделать вывод, что для кодов Рида–Маллера C_1 и C_2 криптосистема $\mathrm{McE}(C_1 \otimes C_2)$ представляется стойкой к структурным атакам на ключ уже при $n_i \geq 8$, где n_i – длина кода C_i, $i = 1, 2$.

Список литературы / References

[1] Shor P. W., “Algorithms for quantum computation: Discrete logarithms and factoring”, *Proceedings 35th Annual Symposium on Foundations of Computer Science*, IEEE Computer Society Press, 1994, 124–134.

[2] Sendrier N., Tillich J.-P., “Code-Based Cryptography: New Security Solutions Against a Quantum Adversary”, *ERCIM News, ERCIM, 2016, Special Theme Cybersecurity (106). hal-01410068.*

[3] McEliece R. J., “A Public-Key Cryptosystem Based on Algebraic Coding Theory”, *JPL Deep Space Network Progress Report*, 1978, № 42, 114–116.

[4] Niederreiter H., “Knapsack-Type Cryptosystem and Algebraic Coding Theory”, *Probl. Control and Inform. Theory*, **15** (1986), 94–34.

[5] Gabidulin E. M. et al., “Ideals Over a Non-Commutative Ring and Their Application in Cryptology”, *Advances in Cryptology–EUROCRYPT’91 / Ed. by D. W. Davies. Lect. Notes in Comp. Sci.*, **547** (1991), 482–489.

[6] Сидельников В. М., “Открытое шифрование на основе двоичных кодов Рида–Маллера”, *Дискретная математика*, **6**:2 (1994), 3–20; [Sidel’nikov V. M., “Open coding based on Reed–Muller binary codes”, *Diskr. Mat.*, **6**:2 (1994), 3–20, (in Russian).]

[7] Сидельников В. М., Шестаков С. О., “О системе шифрования, основанной на обобщенных кодах Рида–Соломона”, *Дискретная математика*, **3**:3 (1992), 57–63; [Sidel’nikov V. M., Shestakov S. O., “O sisteme shifrovanija, osnovannoj pa obobshhennyh kodah Rida–Solomona”, *Diskr. Mat.*, **3**:3 (1992), 57–63, (in Russian).]

[8] Деундяк В. М. и др., “Модификация криптоаналитического алгоритма Сидельникова–Шестакова для обобщенных кодов Рида–Соломона и ее программная реализация”, *Известия высших учебных заведений. Северо-Кавказский регион. Технические науки*, 2006, № 4, 15–20; [Deundyak V. M. et al., “Modifikatsiya kriptoanaliticheskogo algoritma Sidel’nikova–Shestakova dlya obobshchennykh kodov Rida-Solomona i ee programmnaya realizatsiya”, *Izvestiya vysshikh uchebnykh zavedeniy. Severo-Kavkazskiy region. Tekhnicheskie nauki*, 2006, № 4, 15–20, (in Russian).]

[9] Overbeck R., “Structural Attacks for Public Key Cryptosystems based on Gabidulin Codes”, *Journal of Cryptology*, **21**:2 (2008), 280–301.

[10] Minder L., Shokrollahi A., “Cryptanalysis of the Sidelnikov cryptosystem”, *Lecture Notes in Computer Science*, **4515** (2007), 347–360.

[11] Чижов И. И., Бородин М. А., “Эффективная атака на криптосистему Мак-Элиса, построенную на основе кодов Рида–Маллера”, *Дискрет. матем.*, **26**:1 (2014), 10–20; [Chizhov I. I., Borodin M. A., “Jeffektivnaja ataka na kriptosistemu Mak-Jelisa, postroennuju na osnove kodov Rida–Mallera”, *Diskr. Mat.*, **26**:1 (2014), 10–20, (in Russian).]

[12] Деундяк В. М., Косолапов Ю. В., “Криптосистема на индуцированных групповых кодах”, *Модел. и анализ информ. систем*, **23**:2 (2016), 137–152; [Deundyak V. M., Kosolapov Yu. V., “Cryptosystem Based on Induced Group Codes”, *Modeling and Analysis of Information Systems*, **23**:2 (2016), 137–152, (in Russian).]

[13] Деундяк В. М., Косолапов Ю. В., "Алгоритмы для мажоритарного декодирования групповых кодов", *Модел. и анализ информ. систем*, **22**:4 (2015), 464–482; [Deundyak V. M., Kosolapov Yu. V., "Algorithms for Majority Decoding of Group Codes", *Modeling and Analysis of Information Systems*, **22**:4 (2015), 464–482, (in Russian).]

[14] Циммерман К.-Х., *Методы теории модулярных представлений в алгебраической теории кодирования*, МЦНМО, М., 2011; [Tsimmerman K.-Kh., *Metody teorii modulyarnykh predstavleniy v algebraicheskoy teorii kodirovaniya*, MTsNMO, M., 2011, (in Russian).]

[15] Curtis C. W., Reiner I., *Representation Theory of Finite Groups and Associative Algebras*, Intersclence Publishers, New York, 1962.

[16] Lenstra A. K., Verheul E. R., "Selecting Cryptographic Key Sizes", *Journal of Cryptology*, **14** (2001), 255–293.

Deundyak V. M., Kosolapov Y. V., Lelyuk E. A., "Decoding the Tensor Product of MLD Codes and Applications for Code Cryptosystems", *Modeling and Analysis of Information Systems*, **24**:2 (2017), 239–252.

DOI: 10.18255/1818-1015-2017-2-239-252

Abstract. For the practical application of code cryptosystems such as McEliece, it is necessary that the code used in the cryptosystem should have a fast decoding algorithm. On the other hand, the code used must be such that finding a secret key from a known public key would be impractical with a relatively small key size. In this connection, in the present paper it is proposed to use the tensor product $C_1 \otimes C_2$ of group MLD codes C_1 and C_2 in a McEliece-type cryptosystem. The algebraic structure of the code $C_1 \otimes C_2$ in the general case differs from the structure of the codes C_1 and C_2, so it is possible to build stable cryptosystems of the McEliece type even on the basis of codes C_i for which successful attacks on the key are known. However, in this way there is a problem of decoding the code $C_1 \otimes C_2$. The main result of this paper is the construction and justification of a set of fast algorithms needed for decoding this code. The process of constructing the decoder relies heavily on the group properties of the code $C_1 \otimes C_2$. As an application, the McEliece-type cryptosystem is constructed on the code $C_1 \otimes C_2$ and an estimate is given of its resistance to attack on the key under the assumption that for code cryptosystems on codes C_i an effective attack on the key is possible. The results obtained are numerically illustrated in the case when C_1, C_2 are Reed–Muller–Berman codes for which the corresponding code cryptosystem was hacked by L. Minder and A. Shokrollahi (2007).

Keywords: majority decoder, Reed–Muller–Berman codes, tensor product codes

About the authors:
Deundyak Vladimir Mikhailovich, orcid.org/0000-0001-8258-2419, PhD,
FGNU NII "Specvuzavtomatika",
51 Gazetniy lane, Rostov-on-Don 344002, Russia
South Federal University,
105/42 Bolshaya Sadovaya Str., Rostov-on-Don 344006, Russia, e-mail: vl.deundyak@gmail.com,

Kosolapov Yury Vladimirovich, orcid.org/0000-0002-1491-524X, PhD,
South Federal University,
105/42 Bolshaya Sadovaya Str., Rostov-on-Don 344006, Russia, e-mail: itaim@mail.ru,

Leluk Evgeniy Andreevich, orcid.org/0000-0001-6560-2561,
South Federal University,
105/42 Bolshaya Sadovaya Str., Rostov-on-Don 344006, Russia, e-mail: lelukevgeniy@mail.ru