Computing, Telecommunications and Control

Vol. 13, no. 2
2020

COMPUTING, TELECOMMUNICATIONS AND CONTROL

Информатика, телекоммуникации и управление

Том 13, № 2
2020

ИНФОРМАТИКА, ТЕЛЕКОММУНИКАЦИИ И УПРАВЛЕНИЕ

Contents

Information Technologies

Circuits and Systems for Receiving, Transmitting, and Signal Processing

Software of Computer, Telecommunications and Control Systems

System Analysis and Control

Содержание

DATA PROCESSING BY END DEVICES IN IoT SYSTEMS

A.Yu. Anufrienko

National Research University Higher School of Economics,
Moscow, Russian Federation

This paper presents a correlation method for processing data on end devices and reducing the amount of data transmitted over the network. Instead of expensive and complex network devices, developers can use cheap and proven low-speed Internet of Things (ZigBee, NB IoT, BLE) solutions for data transfer. The novelty lies in one of the features of this approach: the use of components for analysis, rather than a complete copy of the signals, as well as processing directly on the sensor. The advantage of this approach allows you to reduce the number of operations and complexity of implementation, in contrast to other methods focused on the cloud computing paradigm. We provide results for correlation values and the number of logical elements (LE) when implemented on the FPGA, depending on the number of elements in the correlator. This allows to maintain a balance between the required calculation accuracy and spent hardware resources, as well as to simplify the end device.

Keywords: Internet of Things, industrial IoT, correlation, FPGA, matched filter, autocorrelation.

ОБРАБОТКА ДАННЫХ КОНЕЧНЫМИ УСТРОЙСТВАМИ В СИСТЕМАХ ИНТЕРНЕТА ВЕЩЕЙ

А.Ю. Ануфриенко

Национальный исследовательский университет «Высшая школа экономики»,
Москва, Российская Федерация

Представлен корреляционный метод для обработки данных на конечных устройствах и сокращения объема передаваемых по сети данных. Вместо дорогих и сложных сетевых устройств разработчики могут применять дешевые и проверенные низкоскоростные решения Интернета вещей (ZigBee, NB IoT, BLE) для передачи данных. Новизна состоит в одной из особенностей этого подхода — использовании для анализа не полной копии сигналов, а их компонентов, а также обработке непосредственно на сенсоре. Данный подход позволяет уменьшить количество операций и сложность реализации в отличие от иных методов, ориентированных на парадигму облачных вычислений. Приведены результаты для значений корреляции и количества логических элементов (LE) при реализации на ПЛИС в зависимости от количества элементов в корреляторе. Это позволяет соблюдать баланс между требуемой точностью расчета и затраченными аппаратными ресурсами, а также упростить конечное устройство.

Ключевые слова: Интернет вещей, промышленный Интернет вещей, корреляция, ПЛИС, согласованный фильтр, автокорреляция.

Introduction

Data are generating by End devices in Internet of Things systems: mainly group of sensors, social media, and applications. This massive data generation results in ''big data'', but not all kinds of data are valuable. Generally, the structure of IoT consists of five layers: Perception Layer, Network Layer, Middleware Layer, Application Layer, and Business Layer. Some of the Internet of Things architectures targeted to cloud computing at the center and a model of end-to-end interaction among various stakeholders in a cloud-centric IoT approach [1]. Cloud computing frees the enterprise and the end user from the specification of many details. This feature becomes a problem for latency-sensitive (industrial) applications, which require minimized delay. Fog computing extends the cloud computing paradigm to the edge of the network. The Fog vision was conceived to address applications and services that do not fit the paradigm of the Cloud well [3].

Typical architecture and components of IoT systems are presented in [1−4]. The systems include modules consisting of sensors, actuators and modems − devices that generate and transmit data. A number of sensors read and report the status of monitored objects. Industrial equipment may have thousands of points for data generation. The module may also have actuators for affecting the logical state of the tool. Modems transmit data to the next level − gate. The gate is usually a hardware component that interacts with a number of modules. The gate also interacts with a platform where the data are saved, processed and provided to end users. The platform (web-based platform) has a number of core components like storage systems, databases, AI and BI tools and an app engine support. So, one of specific devices in the system is the module because the quality of the final results and big data are depending on this device. Typical case for IoT system is transmitting data from modules to an IoT platform (cloud) as is, and deep processing with BI and AI tools.

The volume of generated data is often large, so its transfer to the cloud is limited by the network bandwidth. In industrial IoT applications 100 % of data should be analyzed, but not 100 % of the data should be saved. In addition, there are a lot of other applications (connected cars, smart city, assessments) when several groups of sensors are used in the tests or operations. In aviation, an aircraft engine can have as many as 250 sensors. A twin-engine aircraft on a 12-hour flight can produce up to 844 TB of data [5]. Widely used IoT wireless technologies have typical throughput of 10−250 kbps, and end devices may be autonomous (have an autonomous power supply) and low powered. In addition, there are not enough storage systems for terabytes and petabytes of raw data. This example demonstrates the complexity of using cloud-oriented approaches to analyze high-speed data streams.

The purpose of this paper is to review the existing data processing methods (cloud-edge) and research required computing resources and correlation efficiency depending on the complexity of data processing when processing on devices.

Clouds and endpoint architectures

One of the most commonly used approaches for IoT systems is the sampling rate adaptation [6−9]. A sampling rate is a rate at which a new sample is taken from a continuous signal provided by the sensor board. This rate can be adapted according to the input acquired from the monitoring area. If no significant change is noticed for a certain period of time, the sampling rate could be reduced for the upcoming period, and in contrast, if an event is detected, the sampling rate is increased. This sampling rate adaptation is based on event detection [7]. Data reduction approaches focus solely on reducing the number of transmissions while maintaining a fixed sampling rate [9]. The most popular of them all is the dual prediction scheme [10]. A prediction model capable of forecasting future values is trained and shared between the source and the destination, thus enabling the source sensor node to transmit only the samples that do not match the predicted value.

Another variation is a spatial-temporal correlation based approach for sampling and transmission rate adaptation in cluster-based sensor networks [11]. The correlation between sensor nodes and the new sampling rates of each sensor is calculated. This approach does not require any algorithm to be implemented on the sensor level, the only task performed by the sensors consists exclusively in sampling and transmission. All the work is done on the Cluster-Head (CH) level, where at the end of each round (duration predefined by the user) the CH runs an algorithm that finds the spatial correlation among the data reported by the sensors belonging to the same cluster. Then, it transmits to one of them its new sampling rate for the next round according to its level of correlation with other neighboring sensors in the cluster. The sampling rate scheduling follows a strict protocol that keeps the sampling rate of the sensors showing high correlation with a large number of nodes at an optimal maximum level [11].

In paper [12], the authors propose to capture such sensor data correlation changes to improve the performance of IoT (Internet of Things) equipment for anomaly detection. In a feature selection method, first cluster correlated sensors together to recognize the duplicated deployed sensors according to sensor data correlations, and monitor the data correlation changes in real time to select the sensors with correlation changes as the representative features for anomaly detection. Curve alignment and dynamic time warping (DTW) [13] are methods used for measuring similarity between two time series (data sequences). However, DTW methods do not assume a consistent time lag, and calculate an optimal matching between two given time series with certain restrictions to maximize a measure of their similarity [12]. But these methods involve working with big data at the cloud side.

There are two key issues:
1) Limited IoT network throughput and large amount of data.
2) Limited calculation resources near the sensors.

The general task for an IoT system is a reliable transmission of the data from the sensors to the platform for further analysis and visualization as Fig. 1a shows. As already mentioned, not all data should be transferred and stored for subsequent processing. For deep and precision analyzing, critical modes possess the utmost importance, especially at the assessment stage.
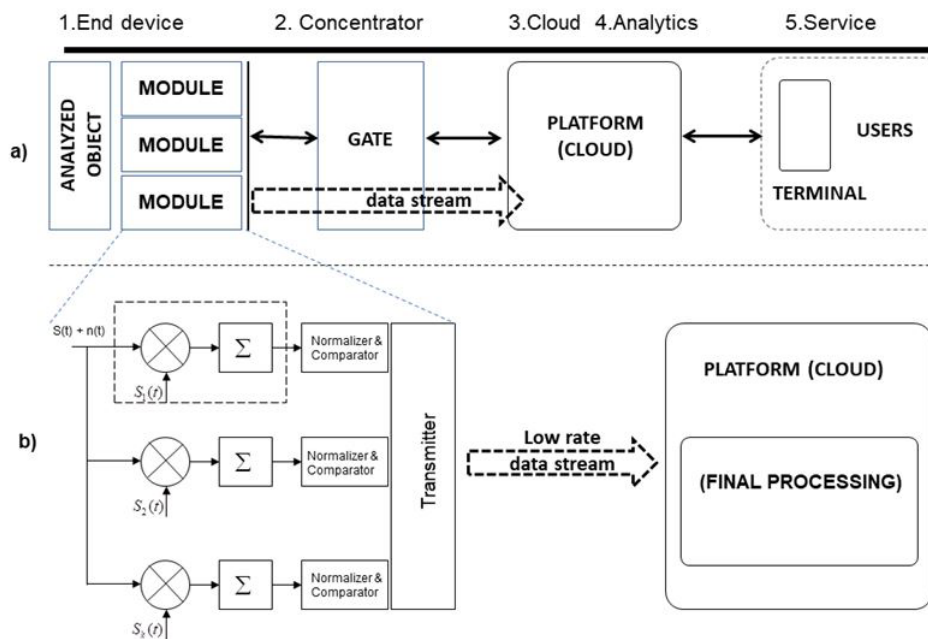
Fig. 1. Typical architecture of IoT systems (a) and simplified block diagram
for a matched filter in a module and low rate data transmission to a cloud (b)

Design and results

To achieve the required results, we could use correlation methods. For many years, correlation methods have been applied in radar and sonar systems for range and position finding in which transmitted and reflected waveforms are compared. In robotic vision, they are used for remote sensing by satellite in which data from different images are compared. One of the applications of correlation is correlation detection implemented by the matched filter, which maximizes S/N ratio at its output. And output result of the matched filter is the autocorrelation at lag zero of input signal and its locally saved copy. But in IoT systems, we operate only with random signals, opposite to radar applications. The cross-correlation [14] between two digital sequences, each containing N data and normalized to the number of samples might be written as:

$$r_{12}(j) = \frac{1}{N}\sum_{n=0}^{N-1} x_1(n) * x_2(n+j), \tag{1}$$

where, the correlation should be calculated with lags. In case when sequence $x_1(n) = x_2(n)$, the process is known as autocorrelation and can be written as:

$$r_{11}(0) = \frac{1}{N}\sum_{n=0}^{N-1} x_1^2(n) = S, \tag{2}$$

where S is normalized energy of signal.

The cross-correlation values computed according to the above equations depend on the absolute values of the data. But it is often necessary to measure cross-correlations in a fixed range $[-1; +1]$. This can be achieved by normalizing the values by an amount depending on the energy content of the data. And the normalized expression for r_{12} becomes:

$$\rho_{12}(j) = \frac{r_{12}(j)}{\frac{1}{N}\sqrt{[\sum_{n=0}^{N-1} x_1^2(n) * \sum_{n=0}^{N-1} x_2^2(n)]}}, \tag{3}$$

where "+1" means complete coincidence (100 % correlation). Despite the fact that the result in the range $[-1; +1]$ is convenient for understanding and analysis, the computational complexity of the denominator (3) is high and requires relatively large computational resources, especially the division operation.

In this paper, we analyze required computing resources and correlation efficiency depending on the complexity of the filter. Matched filters are detecting signals by comparing (determining the correlation) a known signal or pattern with a received signal. So, the number of samples of a known signal (saved copy) defines the number of taps of the matched filter. On-sensor processing concept means a combination of sensor and processor functions in a single device (System-on-chip). The module consists of a sensor, an analog-to-digital converter (ADC) and a fast processor. There are several types of devices suitable for the prototyping tasks − microcontrollers (MCU), digital signal processors (DSP) and field programmable gate arrays (FPGA). Most microcontrollers have a built-in sensors and ADC, but the operation frequency of the MCU is limited. In parallel processing, when a matched filter stores a set of local signals, implementation on the DSP is not the optimal solution. FPGA devices are more optimal for fast parallel processing in case of a matched filter. Cyclone IV FPGA family [16] was used as a base for system prototyping. The models were implemented with MATLAB. FPGA implementation used Verilog HDL. As it was said earlier, Signal-to-Noise ratio at filter's output depends on the quality of a stored copy of a signal. In a digital
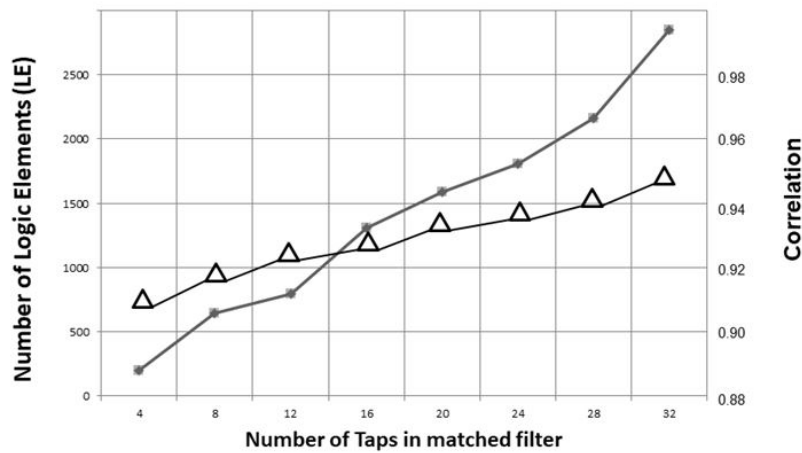
Fig. 2. Correlation values and number of logic elements (LE)
depending on the number of taps in the correlator
(△) − correlation; (♦) − N of LE

system, it means a number of samples of one signal. And the number of samples defines the number of taps in the filter. We used triangular waveforms described by 4, 8, 12, 16, 20, 24, 28 and 32 samples in the experiment. Thus, eight 14-bit matched filters were implemented in the FPGA in accordance with the models made in MATLAB. The use of FPGA resources is analyzed only for multipliers and integrators, shown in Fig. 1b. The amount of logical resources required for the division and square root operations is constant. The results of modeling and implementation are shown in Fig. 2.

199 LEs, 63 registers are required for the 4-tap filter, and a correlation value of 0.91 is reached. And 2646 LEs, 477 registers are required for the 32-tap filter, and a correlation value of 0.948 is reached. In case we need to analyze 100 types of signals in a data stream, 19 900 LEs will be necessary in the first case and 264 600 LEs in the second. Of course, operation frequency will be lower in the second case. Fmax for Cyclone IV is equal to 133 MHz, and the presented method gives a delay of 2 clocks, which is unattainable for MCUs and DSPs, and with parallel processing on the FPGA the performance gain will be more significant. The cost of an FPGA device suitable for implementing 100 32-tap filters can be 10 times higher than that of a simple FPGA device. For ASIC implementation, the number of gates is also crucial.

Conclusion

The key results presented in the article are summarizing fast data processing based on correlation with orientation to FPGA/ASIC. Correlation processing allows to reduce the amount of data transmitting from a sensor to the cloud and to simplify the IoT network architecture. Sensor-based data processing has more advantages than the cloud computing approach, where less than 100 % of the data is required for transmission, storage, and analysis. The applications are especially important in industrial systems (industrial IoT). Due to the large number of multiplications and divisions, correlators require a large amount of hardware resources.

Simulation results show the effectiveness of event detection. The dependence of the required hardware resources of the FPGA on the correlation value increases non-linearly. As the number of taps increases, the performance of the system (Fmax) decreases, so it is important to maintain a balance between accuracy and resource consumption. However, instead of expensive and complex network devices (Fog, Edge, Cloud computing), engineers can use cheaper IoT solutions. In future designs, it is more promising to use SoC solutions that include sensors, ADCS, microcontrollers and logic cores.

REFERENCES

1. **Khan R., Khan S.U., Zaheer R., Khan S.** Future internet: The Internet of Things architecture, possible applications and key challenges. *Proceedings of the 10th International Conference on Frontiers of Information Technology (FIT '12)*, 2012, Pp. 257−260.

2. **Weyrich M., Ebert C.** Reference architectures for the Internet of Things. *IEEE Software*, 2016, Vol. 33, No. 1, Pp. 112−116.

3. **Bonomi F., Milito R., Natarajan P., Zhu J.** Fog computing: a platform for Internet of Things and analytics. *Big Data and Internet of Things: A Road Map for Smart Environments*. Springer, Berlin, Germany, 2014, Pp. 169−186.

4. **Marjani M., et al.** Big IoT data analytics: Architecture, opportunities, and open research challenges. *IEEE Access*, 2017, Vol. 5, Pp. 5247−5261.

5. Engines in the Data Cloud. Available: *https://www.digitalcreed.in/engines-data-cloud/, April, 10, 2018.*

6. **Bhuiyan M.Z.A., Wu J., Wang G., Wang T., Hassan M.M.** e-sampling: Event-sensitive autonomous adaptive sensing and low-cost monitoring in networked sensing systems. *ACM Trans. Auton. Adapt. Syst.*, 2017, Vol. 12.

7. **Harb H., Makhoul A.** Energy-efficient sensor data collection approach for industrial process monitoring. *IEEE Trans. Ind. Informat.*, 2018, Vol. 14, No. 2, Pp. 661−672.

8. **Tayeh G.B., Makhoul A., Laiymani D., Demerjian J.** A distributed real-time data prediction and adaptive sensing approach for wireless sensor networks. *Pervasive Mobile Comput.*, 2018, Vol. 49, Pp. 62−75.

9. **Tayeh G.B., Makhoul A., Demerjian J., Laiymani D.** A new autonomous data transmission reduction method for wireless sensors networks. *Proceedings of the IEEE Middle East North Afr. Commun. Conf. (MENA-COMM)*, 2018, Pp. 1−6.

10. **Braten A.E., Kraemer F.A., Palma D.** Adaptive, correlation-based training data selection for IoT device management. *Proceedings of the 6th International Conference on Internet of Things: Systems, Management and Security (IOTSMS)*, Granada, Spain, 2019, Pp. 169−176.

11. **Tayeh G.B., Makhoul A., Perera C., Demerjian J.** A spatial-temporal correlation approach for data reduction in cluster-based sensor networks. *IEEE Access*, 2019, Vol. 7, Pp. 50669−50680.

12. **Su S., Sun Y., Gao X., Qiu J., Tian Z.** A correlation-change based feature selection method for IoT equipment anomaly detection. *Applied Sciences*, 2019, Vol. 9(3), P. 437.

13. **Kim S., Lee H., Ko H., Jeong S., Byun H., Oh K.** Pattern matching trading system based on the Dynamic TimeWarping Algorithm. *Sustainability*, 2018, No. 10, P. 4641.

14. **Ifeachor E., Jervis B.** *Digital signal processing: A practical approach*. Hardcover, 2nd ed., USA: Prentice Hall, 2001, Pp. 184−245.

15. **Oppenheim A.V., Schafer R.W., Buck J.R.** *Discrete-time signal processing*. Hardcover, 2nd ed., USA: Prentice Hall, 1998. Pp. 746−753.

16. Cyclone IV Device Handbook, 490 p. Available: *https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/hb/cyclone-iv/cyclone4-handbook.pdf.* (Accessed: March 2016).

СПИСОК ЛИТЕРАТУРЫ

1. **Khan R., Khan S.U., Zaheer R., Khan S.** Future internet: The Internet of Things architecture, possible applications and key challenges // Proc. of the 10th Internat. Conf. on Frontiers of Information Technology. 2012. Pp. 257−260.

2. **Weyrich M., Ebert C.** Reference architectures for the Internet of Things // IEEE Software. 2016. Vol. 33. No. 1. Pp. 112−116.

3. **Bonomi F., Milito R., Natarajan P., Zhu J.** Fog computing: a platform for Internet of Things and analytics // Big Data and Internet of Things: A Road Map for Smart Environments. Springer, Berlin, Germany, 2014. Pp. 169−186.

4. **Marjani M., et al.** Big IoT data analytics: Architecture, opportunities, and open research challenges // IEEE Access. 2017. Vol. 5. Pp. 5247−5261.

5. Engines in the Data Cloud // URL: https://www.digitalcreed.in/engines-data-cloud/, April, 10, 2018.

6. **Bhuiyan M.Z.A., Wu J., Wang G., Wang T., Hassan M.M.** e-sampling: Event-sensitive autonomous adaptive sensing and low-cost monitoring in networked sensing systems // ACM Trans. Auton. Adapt. Syst. 2017. Vol. 12.

7. **Harb H., Makhoul A.** Energy-efficient sensor data collection approach for industrial process monitoring // IEEE Trans. Ind. Informat., 2018, Vol. 14, No. 2, Pp. 661−672.

8. **Tayeh G.B., Makhoul A., Laiymani D., Demerjian J.** A distributed real-time data prediction and adaptive sensing approach for wireless sensor networks // Pervasive Mobile Comput. 2018. Vol. 49. Pp. 62−75.

9. **Tayeh G.B., Makhoul A., Demerjian J., Laiymani D.** A new autonomous data transmission reduction method for wireless sensors networks // Proc. of the IEEE Middle East North Afr. Commun. Conf. 2018. Pp. 1−6.

10. **Braten A.E., Kraemer F.A., Palma D.** Adaptive, correlation-based training data selection for IoT device management // Proc. of the 6th Internat. Conf. on Internet of Things: Systems, Management and Security. Granada, Spain, 2019. Pp. 169−176.

11. **Tayeh G.B., Makhoul A., Perera C., Demerjian J.** A spatial-temporal correlation approach for data reduction in cluster-based sensor networks. IEEE Access, 2019, Vol. 7, Pp. 50669−50680.

12. **Su S., Sun Y., Gao X., Qiu J., Tian Z.** A correlation-change based feature selection method for IoT equipment anomaly detection // Applied Sciences. 2019. Vol. 9(3). P. 437.

13. **Kim S., Lee H., Ko H., Jeong S., Byun H., Oh K.** Pattern Matching Trading System Based on the Dynamic Time Warping Algorithm // Sustainability 2018, No. 10, P. 4641.

14. **Ifeachor E., Jervis B.** Digital signal processing: A practical approach. Hardcover, 2nd ed., USA: Prentice Hall, 2001. Pp. 184−245.

15. **Oppenheim A.V., Schafer R.W., Buck J.R.** Discrete-time signal processing. Hardcover, 2nd ed., USA: Prentice Hall, 1998. Pp. 746−753.

16. Cyclone IV Device Handbook, 490 p. // URL: https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/hb/cyclone-iv/cyclone4-handbook.pdf. (Accessed: March 2016).

THE AUTHOR / СВЕДЕНИЯ ОБ АВТОРЕ

Anufrienko Alexander Yu.
Ануфриенко Александр Юрьевич
E-mail: alexanuf@mail.ru

Circuits and Systems for Receiving, Transmitting, and Signal Processing

THE IMPACT OF GNSS SPATIAL SIGNAL PROCESSING ON POSITION AND TIME MEASUREMENTS

P.A. Kudriasheva, A.S. Davydenko

Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation

One of the main research directions in global navigational satellite systems is increasing the intentional interferences resistance of modern navigation receiving equipment. The most effective method is supposed to be the use of spatial filtering techniques on the basis of adaptive antenna arrays (AAA). However, antenna array can bring about additional errors in the navigation and make it impossible to use it for applications requiring accurate positioning and time synchronization. We experimentally compared navigation solutions obtained based on signals from a single antenna and from the output of AAA. The results showed that the use of AAA as the part of navigation receiver might delay 1 pps (pulse per second) signal arrival on the value proportional to the summarized group delay in the digital signal processing block of AAA. Experimental results also showed that AAA could bring error to positioning of the receiver. A few methods were outlined to decrease the influence of AAA on navigation solution.

Keywords: global navigation satellite system, adaptive antenna array, 1 pps-signal, positioning, navigation receiver.

ВЛИЯНИЕ ПРОСТРАНСТВЕННОЙ ОБРАБОТКИ СИГНАЛОВ ГНСС НА НАВИГАЦИОННО-ВРЕМЕННЫЕ ОПРЕДЕЛЕНИЯ

П.А. Кудряшева, А.С. Давыденко

Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация

Повышение помехозащищенности – важная задача, требующая решения в современной приёмной аппаратуре сигналов глобальных навигационных спутниковых систем. Одним из наиболее эффективных методов является пространственная фильтрация на основе адаптивных антенных решеток (ААР). Однако антенная решетка может приводить к появлению дополнительных ошибок в решении навигационной задачи, что делает невозможным использование ААР для приложений, требующих точного позиционирования и точной временной синхронизации. Проведено экспериментальное сравнение навигационных решений, полученных на основе сигналов с одиночной антенны и на основе сигнала на выходе ААР. Результаты показали, что использование ААР в составе приёмной навигационной аппаратуры может привести к задержке выдачи сигнала 1 pps на величину, пропорциональную времени задержки сигнала в цифровой части ААР. Результаты эксперимента также показали, что ААР может привести к появлению ошибки в позиционировании приёмника. Предложены варианты по уменьшению влияния ААР на решение навигационной задачи.

Introduction

Nowadays global navigation satellite systems (GNSS) find a wide range of applications in various fields of science and technology. GNSS allows determining position and speed of objects with high accuracy, to determine angular orientation and provide time synchronization of GNSS consumer equipment.

The main vulnerability of GNSS is caused by GNSS equipment susceptibility to intentional interference (jamming signals). One of the main research directions in GNSS is increasing the intentional interferences resistance of GNSS consumer equipment.

Reliable operation of GNSS receivers in the presence of jamming signals can be maintained by interference filtration techniques (time-frequency, polarization, spatial filtration), but the most effective and universal one is supposed to be spatial filtration on adaptive antenna arrays [1–5]. Spatial filtration technique is based on processing signals received on spaced antenna elements.

An adaptive antenna array (AAA) is a set of antenna elements whose channels gain can be controlled in amplitude and phase, that feature allows to shape desired radiation pattern of the AAA. To suppress interference, it is necessary to generate zeros of the radiation pattern at the interference arrival directions.

AAA research usually evaluates interference suppression performance and little attention is given to the impact of AAA on signals of interest, particularly on GNSS signals. In practice, the use of weighting processing, non-identical frequency characteristics of receiving channels, the use of antenna elements with non-identical radiation patterns can lead to the formation of additional amplitude-phase shift at the AAA output signal [6–8], the shift can introduce additional error in the solution of the navigation problem. Due to this additional error, the range of AAA applications as a part of navigation equipment (that requires high-precision positioning and/or accurate timing synchronization) can be reduced.

In papers [8–12] the influence of AAA on the operation of a GNSS receiver is shown by estimation of intermediate parameters of GNSS signal processing: pseudo-ranges and code or carrier phase biases. These papers do not describe how pseudo-ranges or phase biases could affect positioning or time synchronization pulse generating. In addition, the final result depends on the type of AAA algorithm used. In [13] the measured time delay is achieved in a few decimeters. In [14] after estimating the offset, the receiver offset errors could be compensated either in the navigation processor or in the tracking loop of the GNSS receiver. The simulation demonstrated centimeter-level bias correction accuracy.

The navigation solution can be produced on the basis of code or phase measurements [8–12], but in this work, we pay attention to code measurements.

The purpose of this research is to identify the impact of AAA on the navigation solution by comparing the accuracy of the navigation solution with and without using AAA. As the measure of AAA impact on navigation solution we used the deviation of coordinates in rectangular coordinate system relative to reference point and average time delay of synchronizing 1 pps pulses using AAA instead of a single antenna element for measurements.

Adaptive antenna array

Interference filtration by AAA is based on the principle of spatial selectivity. The main characteristic of AAA is the radiation pattern (RP) — dependence of the AAA gain on signal arrival direction. In order to suppress interference, it is necessary to shape the RP's zeros in the direction of the interference arrival.

Fig. 1 shows the structural diagram of AAA with N antenna elements. Interferences and signals of interest from satellites are received by antenna elements. Signals from each antenna element pass to the radionavigation receiving device, then analog-to-digital conversion (ADC) of the signals occurs, and the further processing is performed with digitized signals.

ADC-block forms $X(k) = \left[x_1(k)\, x_2(k) \ldots x_N(k) \right]^T$ samples of input signals for all AE, digitized input signals are further multiplied by complex-conjugated $W = [w_1, w_2, \ldots, w_N]$ weight coefficients and the obtained products are summed up. The sample of AAA output signal for the kth moment of time is calculated as follows:

$$Y(k) = W^H X(k) = \sum_{n=1}^{N} w_n^* x_n(k),$$

the superscript H denotes Hermitian transpose, asterisk * denotes the complex conjugation.

Further, AAA weight coefficients are calculated on the basis of $X(k)$ and $Y(k)$ samples, AAA weights enable generating the AAA RP for interference suppression. There is a great variety of AAA algorithms based on following criteria: minimum mean square error, minimum output power, maximum SNR at the AAA output, etc. If navigation chips are used as GNSS consumer equipment, the AAA output signal is converted to analog form (DAC) (Fig. 1). The output signal of AAA is free of interference signals and used for calculation of navigation solution at the receiver.

Experimental setup

The purpose of the experiment is to determine AAA impact on navigation solution, evaluate the accuracy of the consumer's position and the accuracy of the moment 1 pps signal arrives from navigation receiver. The structural scheme used for measurements is shown in Fig. 2.

The list of the equipment:

navigation reception antenna L1 GPS/GLONASS Tallysman 33-7972-00-1500 (1 piece);

navigation receivers: u-blox LEA-M8T (2 pieces) (accuracy of positioning − 2.5 m, accuracy of 1 pps-signals delivery ≤ 20 ns);

a sample of a 4-element AAA for GNSS signals;

a two-channel device for recording moments of 1 pps signals arrival from navigation receivers;

PC with installed software for operation with navigation receivers and software to form1 pps-signals records.

A mixture of real satellite signals with AWGN is received on two antenna modules. A single antenna represents the first antenna module and the second is the sample of AAA. AAA is capable of operating in
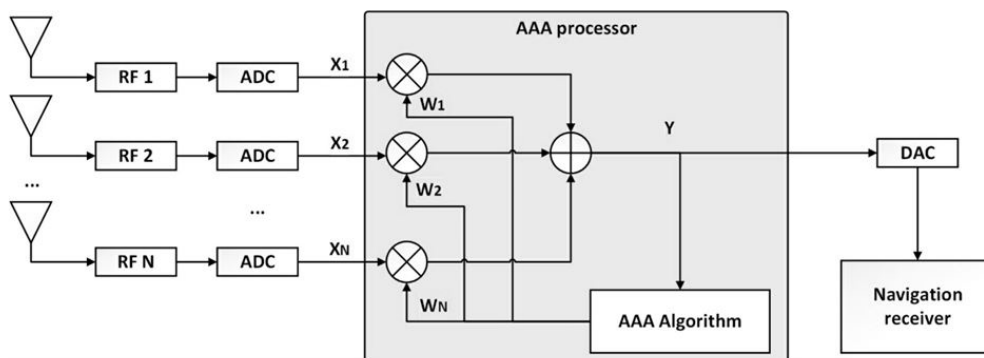
Fig. 1. Structural diagram of AAA with N antenna elements
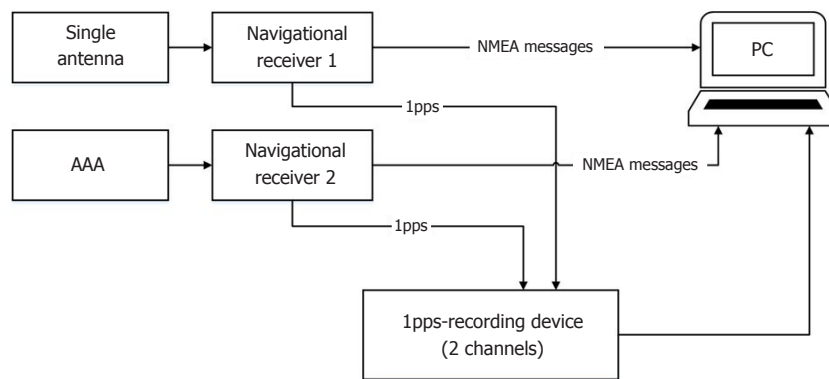
Fig. 2. Block diagram of the measuring equipment

the L1 frequency range of GNSS GPS and GLONASS. Signals from antenna modules are transmitted to inputs of corresponding navigation receivers, where the complete cycle of satellite signals processing is performed, as a result of which the solution of navigation task is evaluated, i.e. position and 1 pps-signal.

Both navigation receivers send NMEA messages to the PC via a serial port once per second and the PC writes them to a text file. Geographical coordinates (latitude, longitude, height) and their corresponding time are extracted from NMEA messages (GGA − Global positioning system fix data) and transformed into rectangular coordinates (x, y, z). Receivers also output a 1 pps signal at 1 Hz. The time of arrival of 1 pps signals is recorded by a two-channel 1 pps-recorder. The 1 pps-recorder contains a 240 MHz reference clock. There is also a counter incrementing every cycle of the reference clock. The second counter fixes moments of 1 pps arrival from a navigation receiver. The second counter increments after 1 pps tag is received and fixes the value until the next 1 pps tag is received. Obtained values of the second counter are recorded into a separate text file with a rate of 2 kHz. Recording is performed simultaneously via two channels from identical navigation receivers. As a result, two-channel record is formed containing arrival moments of the 1 pps signal samples relatively reference 240 MHz clock. Thus, the 1 pps edge is measured with 4 ns precision.

Experimental results

The purpose of the experiment is to compare navigation solution obtained based on signals received at a single antenna element; the measuring device is in the stationary state during measurements.

Comparison of delay of 1 pps signals with AAA relative to 1 pps signals from single antenna is carried out at generation of 1 pps signal on the basis of GPS satellites constellation. The experiment involves comparing the delay of 1 pps without an intentional interference and in the presence of one. However, in the presence of the interference, the navigational receiver is not able to get solution. Therefore, the following sets of records were made to make comparison of the operation navigation receivers with antenna and AAA possible in the presence of interference:

1. All receivers are configured to receive GPS signals. Records are made without intentional interference.

2. The receiver with the single antenna operates on GLONASS signals, the receiver with AAA operated on GPS signals. Records are made without intentional interference.

3. The receiver with the single antenna operates on GLONASS signals, the receiver with AAA operates on GPS signals. Records are made in the presence of 1 MHz wideband interference in the GPS signal band.

Each record set contains: records of NMEA messages from each navigation receiver; a two-channel record 1 pps signals from receivers. Measurements are made under conditions of direct reception of satellite signals during 20 minutes, the rate of navigation solution output − 1 Hz.

All coordinates are measured relatively (X_{ref}, Y_{ref}, Z_{ref}) − the reference point measured with centimeter-accuracy by the Trimble R7 GNSS Receiver. Based on the obtained records sets, we transformed the geodesic coordinates to rectangular and constructed histograms of rectangular coordinates (x, y, z) (Fig. 3−5). We also calculated sample mean and standard deviation of relative coordinates and tabulated the results (Table 1).
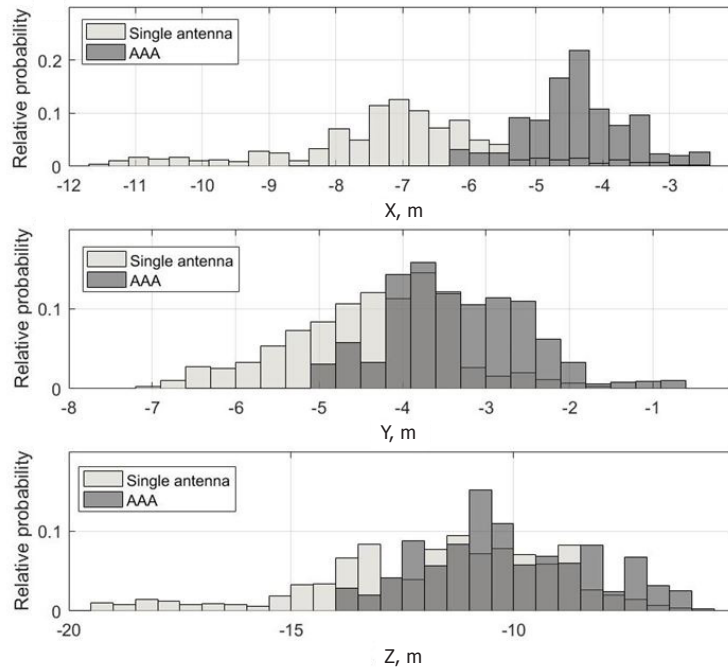
Fig. 3. Histograms of measured coordinates, both receivers operate on GPS, without interference
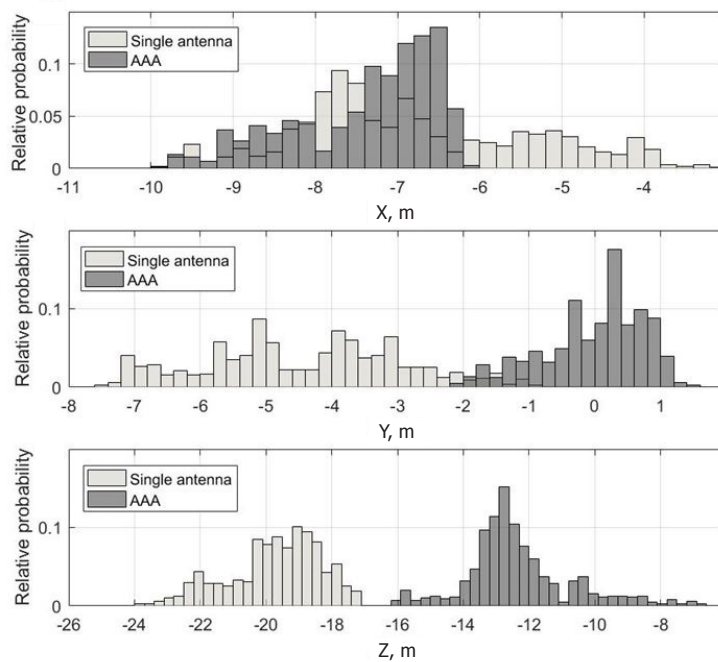
Fig. 4. Histograms of measured coordinates, receiver with single antenna operates
on GLONASS, the other − on GPS, without interference

18
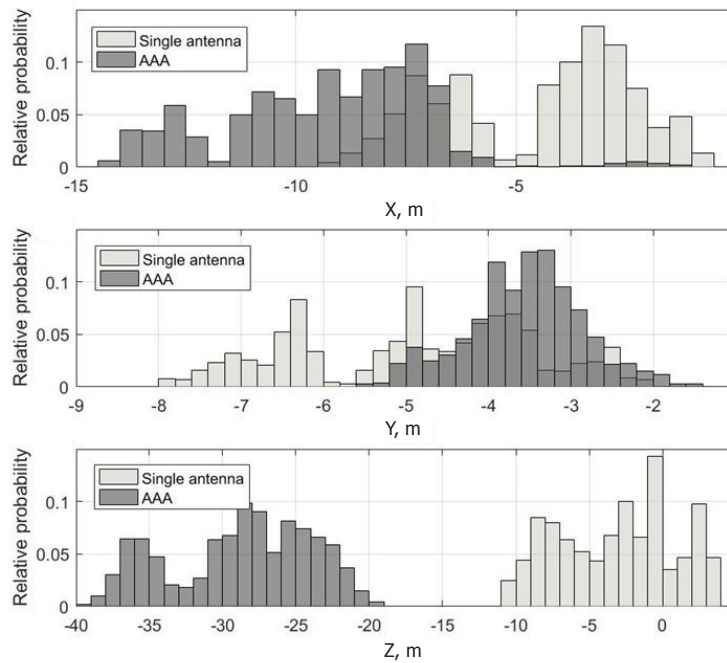
Fig. 5. Histograms of measured coordinates, receiver with single antenna operates
on GLONASS, the other − on GPS, the presence of 1 MHz wideband interference for GPS

T a b l e 1

**Sample mean and standard deviations of measured coordinates
in relation to the reference point**

No. of the records set	Antenna module of receiver	Mean $(X-X_{ref})$, m	Mean $(Y-Y_{ref})$, m	Mean $(Z-Z_{ref})$, m	Std $(X-X_{ref})$, m	Std $(Y-Y_{ref})$, m	Std $(Z-Z_{ref})$, m
1	Single antenna	−7.13	−4.38	−11.73	1.59	0.97	2.63
	AAA	−4.39	−3.34	−10.09	0.79	0.84	1.88
2	Single antenna	−6.83	−4.44	−19.73	1.46	1.50	1.42
	AAA	−7.38	−0.01	−12.36	0.88	0.76	1.62
3	Single antenna	−4.49	−4.87	−3.05	2.15	1.42	4.00
	AAA	−9.26	−3.56	−28.76	2.34	0.72	4.83

We have estimated the delay of the 1 pps signals introduced by the AAA in relation to 1 pps signals generated from the receiver with the single antenna on the basis of two-channel 1 pps signal records. Estimated delays are summarized at Table 2 and are equivalent to the time delay introduced by the AAA.

Using the data from Table 2, we can estimate the delay Δt_{GPS} introduced by the AAA generating a 1 pps signal on GPS signals in the presence of interference based on the estimate of the 1 pps signal delay for the third set of records (during the third recording, a single antenna receiver generates a 1 pps signal via the constellation GLONASS):

$$\delta = \Delta t_1 - \Delta t_2,$$

$$\Delta t_{\text{GPS}} = \Delta t_3 + \delta,$$

where $\Delta t_{1,2,3}$ – estimates of 1 pps signal delay for the first, the second or the third set of records (from Table 2) determined for AAA; δ – difference of 1 pps signal delays caused by operation on different GNSS; Δt_{GPS} – the estimate of the 1 pps delay introduced by AAA signal using GPS constellation in the presence of intended interference. Table 3 contains AAA estimated delays for 1 pps signal without and with the interference effect. Without interference, the AAA sample introduces a delay of 22.2 μs. In the presence of wideband interference, the 1 pps signal is delayed by 22.145 μs.

T a b l e 2

Estimated offset of the 1 pps signal when using AAA in relation to the 1 pps signal from a single antenna

Set of records	Initial conditions	Histogram with coordinates	Estimated AAA delay, μs
1	All receivers are configured to receive GPS signals. Records are made without intentional interference	Fig. 3	22.200
2	The receiver with the single antenna operates on GLONASS signals, the receiver with AAA operates on GPS signals. Records are made without intentional interference	Fig. 3	22.157
3	The receiver with the single antenna operates on GLONASS signals, the receiver with AAA operates on GPS signals. Records are made in the presence of 1 MHz wideband interference in the GPS signal band	Fig. 3	22.102

The delay introduced into the signal by AAA is supposed to be constant and can be attributed to the structure of analog and digital parts of AAA, i.e. signals received at AAA antenna elements are delayed within analog paths of RF-block. Fig. 6 shows a structure diagram of a digital signal processing block for one of the AAA channels. The main contribution to the delay of AAA signals (Table 3) is made by the group delays of digital filters used for signal resampling (down- and upsampling) and filters for correction of phase frequency characteristics of AAA receiving channels; the delay can also be formed by the AAA algorithm (the use of spatial-time processing additionally requires delay taps in each AAA channel).

T a b l e 3

Comparison of 1 pps delay for AAA using GPS constellation

	Estimated AAA delay for GPS constellation, μs
no interference	22.200
1 MHz wideband interference	22.145

Conclusion

By comparing the accuracy of the evaluated navigation solution without and with the use of AAA, we found that the AAA sample delays the output of the 1 pps signal by 22.2 μs in relation to the 1 pps signal from a single antenna element.

The results of coordinate measurements (Table 1) show that without interference the sample mean and standard deviation of the measured coordinates with the single antenna and the AAA slightly differ from each other. In the presence of wideband interference (record set 3), the standard deviation of the vertical
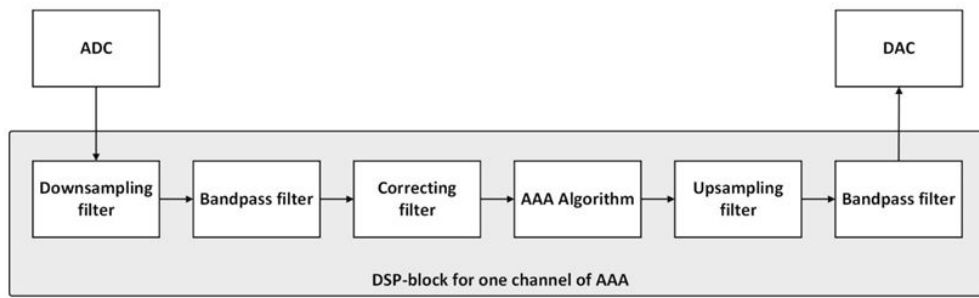
Fig. 6. Structure diagram of a digital signal processing block for one of the AAA channels

coordinate component $std(Z - Z_{on})$ increases by a factor of 2 compared with measurements without the interference. Sample mean coordinate $mean(Z - Z_{on})$ with the single antenna is $-$ 3.05 m, and $mean(Z - Z_{on})$ with AAA is $-$ 28.76 m. In case of interference effect, the use of AAA made the displacement of the measured Z coordinate, at the same time $std(Z - Z_{on})$ with single antenna and with AAA did not differ.

The results showed that the use of AAA as a part of GNSS receiving equipment made an impact on the navigation solution. The AAA influence on the time component can be compensated by the configuration of navigation receiver, the output 1 pps signal delays according to the measured delay value from Table 3. Influence of AAA on navigation parameters, such as, coordinates and speed, can be reduced only by reduction of AAA group delay or by taking into account AAA characteristics for navigation solution calculation.

REFERENCES

1. **Tyapkin V.N., Garin Ye.N.** *Metody opredeleniya navigatsionnykh parametrov podvizhnykh sredstv s ispolzovaniyem sputnikovoy radionavigatsionnoy sistemy GLONASS* [*Methods of determining navigation parameters of mobile means using the GLONASS satellite radio navigation system*]. Krasnoyarsk: Sibirskiy Federalnyy Universitet Publ., 2012. (rus)

2. **Perova A.I., Kharisova V.N. eds.** *GLONASS. Printsipy postroyeniya i funktsionirovaniya* [*GLONASS. Principles of construction and functioning*]. Moscow: Radiotekhnika Publ., 2010. (rus)

3. **Nemov A.V.** *Tsifrovyye antennyye reshetki: novoye kachestvo sputnikovykh radionavigatsionnykh sistem* [*Digital antenna arrays: new quality of radionavigation satellite systems*]. St. Petersburg: SPbGETU "LETI" Publ., 2014. 159 p. (rus)

4. **Yaskin Yu.S., Kharisov V.N., Yefimenko V.S., Boyko S.N., Bystrakov S.G., Pastukhov A.V., Savelyev S.A.** Kharakteristiki podavleniya pomekh v pervom obraztse pomekhoustoychivoy apparatury potrebiteley SRNS GLONASS/GPS s adaptivnoy antennoy reshetkoy [Characteristics of interference suppression by the first sample of the noise immunity equipment of GNSS GLONASS/GPS with an adaptive antenna lattice]. *Radiotekhnika*, 2010, No. 7, Pp. 127−136. (rus)

5. **Kharisov V.N., Yefimenko V.S., Oganesyan A.A., Pastukhov A.V., Pavlov V.S., Golovin P.M., Medvedev P.V.** Otsenka kharakteristik podavleniya pomekh priyemnikam GNSS s antennymi reshetkami v realnykh usloviyakh [Estimation of interference suppression capabilities GNSS receivers with antenna arrays in real conditions]. *Radiotekhnika*, 2013, No. 7, Pp. 132−136. (rus)

6. **Tyapkin V.N., Fateyev Yu.L., Sharfunova T.G., Kurnosov A.S., Shtro P.V.** Eksperimentalnyye issledovaniya pogreshnosti izmereniya psevdodalnostey v apparature potrebiteley navigatsionnoy informatsii s fazirovannoy antennoy reshetkoy [Experimental studies of error of measurement of pseudorange in equipment of users of navigation information with phased antenna array]. *Vestnik SibGAU*, 2013, No. 3, Pp. 143−148. (rus)

7. **Sharfunova T.G., Tyapkin V.N., Dmitriyev D.D.** Tochnost izmereniya navigatsionnykh parametrov v navigatsionnoy apparature potrebitelya sputnikovoy radionavigatsionnoy sistemy GLONASS, osnashchennoy

antennoy reshetkoy [Accuracy of measurement of navigation parameters in navigation equipment of the consumer of the GLONASS satellite radio navigation system equipped with antenna array]. *Radiotekhnika*, 2013, No. 6, Pp. 22−25. (rus)

8. **Lu Z., Chen F., Xie Y., Sun Y., Cai H.** High precision pseudo-range measurement in GNSS anti-jamming antenna array processing. *Electronics*, 2020, No. 9.

9. **Vagle Niranjana, et al.** Analysis of multi-antenna GNSS receiver performance under jamming attacks. *Sensors,* Basel, Switzerland, 2016, Vol. 16(11), P. 1937. DOI: https://doi.org/10.3390/s16111937

10. **Cao L., An X., Hong G., Guo B.** Analysis of measurement biases induced by adaptive antenna arrays for GNSS receivers. *Proceedings of the 12ᵗʰ International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery*, 2016, Pp. 1863−1867.

11. **Vagle N., Broumandan A., Jafarnia A., Lachapelle G.** Characterization of GNSS measurement distortions due to antenna array processing in the presence of interference signals. *2014 Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS)*, Corpus Christ, TX, 2014, Pp. 71−80.

12. **McGraw G.A., McDowell C., Kelly J.M.** GPS anti-jam antenna system measurement error characterization and compensation. *Proceedings of the 19ᵗʰ International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2006)*, Fort Worth, TX, Sept. 2006, Pp. 705−714.

13. **Kihoon Lee, Junpyo Park, Jiyun Ltt.** Effects on the positioning accuracy of a GPS receiver with array antenna and time delay compensation for precise anti-jamming. *Transactions of the Japan Society for Aeronautical and Space Sciences*, 2018, Vol. 61, Issue 4, Pp. 171−178.

14. **O'Brien A.J., Gupta I.J.** Mitigation of adaptive antenna induced bias errors in GNSS receivers. *IEEE Transactions on Aerospace and Electronic Systems*, 2011, Vol. 47, No. 1, Pp. 524−538.

СПИСОК ЛИТЕРАТУРЫ

1. **Тяпкин В.Н., Гарин Е.Н.** Методы определения навигационных параметров подвижных средств с использованием спутниковой радионавигационной системы ГЛОНАСС: Монография. Красноярск: Сибирский федеральный университет, 2012.

2. ГЛОНАСС. Принципы построения и функционирования / Под ред. А.И. Перова, В.Н. Харисова. Изд. 4-е, перераб. и доп. М.: Радиотехника, 2010.

3. **Немов А.В.** Цифровые антенные решетки: новое качество спутниковых радионавигационных систем: Монография. СПб.: Изд-во СПбГЭТУ "ЛЭТИ", 2014. 159 с.

4. **Яскин Ю.С., Харисов В.Н., Ефименко В.С., Бойко С.Н., Быстраков С.Г., Пастухов А.В., Савельев С.А.** Характеристики подавления помех в первом образце помехоустойчивой аппаратуры потребителей СРНС ГЛОНАСС/GPS с адаптивной антенной решёткой // Радиотехника (Журнал в журнале). 2010. № 7. С. 127−136.

5. **Харисов В.Н., Ефименко В.С., Оганесян А.А., Пастухов А.В., Павлов В.С., Головин П.М., Медведев П.В.** Оценка характеристик подавления помех приемникам ГНСС с антенными решетками в реальных условиях // Радиотехника. 2013. № 7. С. 132−136.

6. **Тяпкин В.Н., Фатеев Ю.Л., Шарфунова Т.Г., Курносов А.С., Штро П.В.** Экспериментальные исследования погрешности измерения псевдодальностей в аппаратуре потребителей навигационной информации с фазированной антенной решеткой // Вестник СибГАУ. 2013. № 3. С. 143−148.

7. **Шарфунова Т.Г., Тяпкин В.Н., Дмитриев Д.Д.** Точность измерения навигационных параметров в навигационной аппаратуре потребителя спутниковой радионавигационной системы ГЛОНАСС, оснащенной антенной решеткой // Радиотехника. 2013. № 6. С. 22−25.

8. **Lu Z., Chen F., Xie Y., Sun Y., Cai H.** High precision pseudo-range measurement in GNSS anti-jamming antenna array processing // Electronics. 2020. No. 9.

9. **Vagle N., et al.** Analysis of multi-antenna GNSS receiver performance under jamming attacks // Sensors. Basel, Switzerland, 2016. Vol. 16(11). P. 1937. DOI: https://doi.org/10.3390/s16111937

10. **Cao L., An X., Hong G., Guo B.** Analysis of measurement biases induced by adaptive antenna arrays for GNSS receivers // Proc. of the 2016 12th Internat. Conf. on Natural Computation, Fuzzy Systems and Knowledge Discovery. 2016. Pp. 1863−1867

11. **Vagle N., Broumandan A., Jafarnia A., Lachapelle G.** Characterization of GNSS measurement distortions due to antenna array processing in the presence of interference signals // 2014 Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS). Corpus Christ, TX, 2014. Pp. 71−80.

12. **McGraw G.A., McDowell C., Kelly J.M.** GPS anti-jam antenna system measurement error characterization and compensation // Proc. of the 19th Internat. Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2006). Fort Worth, TX, Sept. 2006. Pp. 705−714.

13. **Kihoon Lee, Junpyo Park, Jiyun Ltt.** Effects on the positioning accuracy of a GPS receiver with array antenna and time delay compensation for precise anti-jamming // Transactions of the Japan Society for Aeronautical and Space Sciences. 2018. Vol. 61. Issue 4. Pp. 171−178.

14. **O'Brien A.J., Gupta I.J.** Mitigation of adaptive antenna induced bias errors in GNSS receivers // IEEE Transactions on Aerospace and Electronic Systems. 2011. Vol. 47. No. 1. Pp. 524−538.

Статья поступила в редакцию 11.03.2020.

THE AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Kudriasheva Polina A.
Кудряшева Полина Андреевна
E-mail: kudriasheva.pa@gmail.com

Davydenko Anton S.
Давыденко Антон Сергеевич
E-mail: ammodo@ya.ru

TOP-DOWN DESIGN OF INTEGRATED TRANSCEIVER: PECULIARITY AND TEACHING METHOD USING EDA ADVANCED DESIGN SYSTEM (ADS)

E.V. Balashov

Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation

Requirements for the degree of integration, cost, power consumption of transceivers are constantly increasing. In this regard, at present, transceivers are built based on integrated circuits. For the successful development of an integrated transceivers, it is necessary to use the principles of top-down design and end-to-end design, implemented in EDA. The article examines the features and methods of teaching students to develop integrated transceivers in accordance with the principle of the top-down design. The proposed teaching methodology allows students to learn how to use system and circuit modeling tools using EDA Advanced Design System (ADS) produced by Keysight as an example and to study a design flow that includes the following stages: system-level design, block-level design, schematic-level design and layout level design.

Keywords: receiver, transceiver, top-down design, EDA, ADS.

НИСХОДЯЩЕЕ ПРОЕКТИРОВАНИЕ ИНТЕГРАЛЬНЫХ ПРИЁМНЫХ И ПЕРЕДАЮЩИХ УСТРОЙСТВ: ОСОБЕННОСТИ И МЕТОДИКА ПРЕПОДАВАНИЯ С ИСПОЛЬЗОВАНИЕМ САПР ADVANCED DESIGN SYSTEM (ADS)

Е.В. Балашов

Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация

Требования к степени интеграции, стоимости, энергопотреблению приёмно-передающих устройств постоянно возрастают. В связи с этим в настоящее время приёмные и передающие устройства строятся на основе интегральных схем. Для успешной разработки интегрального приёмно-передающего устройства необходимо при проектировании использовать принципы нисходящего проектирования (top-down design) и сквозного проектирования, реализованные в системах автоматизированного проектирования радиоэлектронных устройств. В статье рассмотрены особенности и методика обучения студентов разработке интегральных приёмных и передающих устройств в соответствии с принципом нисходящего проектирования. Предложенная методика обучения позволяет студентам научиться использовать средства системного и схемотехнического моделирования на примере системы автоматизированного проектирования Advanced Design System (ADS) компании Keysight и освоить маршрут проектирования, включающий следующие этапы: проектирование на уровне системы, проектирование на уровне струк-

турной схемы, проектирование на уровне принципиальной схемы и проектирование на уровне компоновки кристалла.

Ключевые слова: приёмник, передатчик, нисходящее проектирование, САПР, ADS.

Introduction

Modern digital communication systems are built using complex signal processing algorithms, and the complexity of systems increases as new modulation schemes, multiple access protocols and communication standards are introduced [1–5]. Requirements for the integration degree, cost, power consumption and time to market for a transceiver are constantly increasing. In this regard, at present, transceivers are built using integrated circuits (IC). Integrated circuit includes not only functional units and transceiver blocks, but in some cases, transceivers are fully implemented in the form of an integrated circuit.

For the successful development of an integrated transceiver, it is necessary to use modern design principles and tools. This need arises from the fact that the transceivers of any modern digital communication system represent a set of software and hardware, which are developed in close connection with each other [6–11]. Thus, the transceiver consists of a digital part and an analog part. The digital part performs digital signal processing using algorithms specified at the software or hardware level. The analog part implements analog signal processing, including the processing at high frequencies. For example, the requirements for the schematic and parameters of the transceiver blocks of the analog and high-frequency parts follow from the specification and the design features of the whole system. At the same time, in contrast to the design using discrete elements, the cost of an error in the IC transceiver design is high due to the impossibility of adjustments after the manufacture of the IC chip without redesigning and manufacturing the IC. In this regard, at present, the design of transceivers is carried out on the basis of the principle of top-down design [12] and the principle of end-to-end design flow implemented in electronic design automation (EDA), for example, Advanced Design System (ADS) from Keysight Technologies.

The principle of top-down design implies design with sequential simplification of the task by dividing it into several separate subtasks. Initially, the system is described at a high level, and then gradually divided into separate parts with more detailed description until the granularity is sufficient to implement the system at the level of schematic and IC layout, or until the granularity is sufficient for coding. Three levels of design are distinguished: design at the system level, design at the block diagram level, and design at the schematic diagram level. Each design level uses its own methods of modeling the system or device. A fourth design level – design at the chip layout level can be added to these three.

The end-to-end design flow implies the organization of teamwork on a project in a single development environment that supports various design and simulation tools with the transfer of the results of one design stage to the next. At the same time, changes made at any stage should be reflected in all parts of the project at once.

The developers of digital communication systems, transceivers and their individual parts for the successful solution of his task must have not only knowledge sufficient to solve a specific problem, but also an understanding of the entire design route of the transceiver and the place of the task they are solving specifically among other tasks solved by their colleagues.

In this regard, it seems expedient to train engineers and specialists in the field of design of transceivers on the basis of teaching methods involving the principles of top-down and end-to-end design flow. This

will not only create a holistic view of the students about the subject, but also teach them the principles of a modern approach to designing receivers and transmitters.

Peculiarity of integrated transceiver design

The signal processing of digital communication systems is realized using digital or analog signal processing based on hardware and software. The hardware can include the following main components responsible for the signal processing steps:

– digital signal processor (DSP);
– digital signal processing circuits;
– analog-to-digital and digital-to-analog converter;
– analog part of the transceiver;
– high-frequency part of the transceiver.

Various stages of signal processing can be implemented both in the digital and in the analog domain. For example, consider the implementation of a superheterodyne receiver in Fig. 1. The receiver front-end in Fig. 1 consists of band-select filter (BSF), the low-noise amplifier (LNA), mixer (Mixer 1), image-reject filter (IRF) and intermediate-frequency amplifier (IFA). The signal spectrum is first transferred to the intermediate frequency fIF by means of the mixer (Mixer 1), and then, by means of I/Q demodulator, the passband signal is demodulated.

The signal is demodulated in the analog domain using a quadrature demodulator based on two mixers, Mixer 2 and Mixer 3, and low-pass filters (LPF) (Fig. 1 a). Then the signal is amplified by baseband amplifier (BBA) and the analog-to-digital conversion is carried out by means of a digital-to-analog converter (ADC) and further signal processing is realized in the digital domain (DSP). In the second case (Fig. 1 b), signal demodulation is carried out using I/Q demodulator implemented in the digital domain and located after the analog-to-digital converter (ADC). It consists of the two multipliers and the FIR digital filters. Such a receiver is often called a Low-IF Receiver, or Digital IF Receiver.

At present, due to the rapid development of microelectronics, several trends are observed in the design of transceivers. Firstly, the development of integrated technologies, especially CMOS technology,
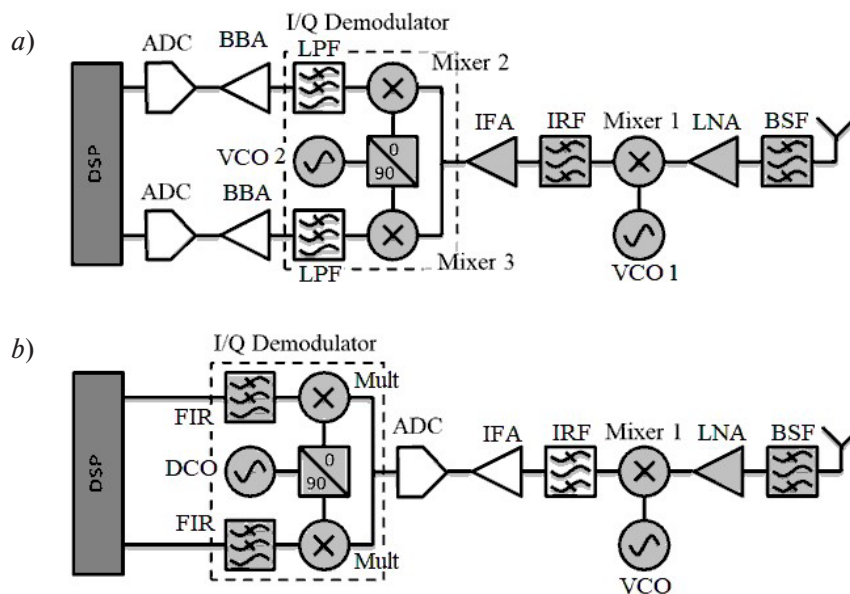
Fig. 1. Block diagram of the receiver with implementation of the demodulator
in the analog domain (*a*), in the digital domain (*b*)

has made it possible to implement the system-on-a-chip (SoC) concept, when both digital, analog and high-frequency parts of the transceiver are integrated on a single chip. Thus, there is a trend towards combining different parts of the receiver on a single integrated circuit chip. Secondly, an increase in the clock frequency of digital circuits made it possible to implement digital signal processing at those frequencies at which previously processing was carried out only in the analog domain. Thirdly, the development of microwave microelectronics has made it possible to use new frequency ranges for data transmission, which were not previously used in consumer electronics. For example, in the frequency range up to 10 GHz, digital signal processing methods have become possible, and in the frequency range up to 100 GHz, personal local networks operate. Fourthly, sophisticated digital modulation schemes are now being used to increase the data rate through the performance of digital circuits.

The complexity of the structure of the transceivers makes it necessary to use special approaches, software tools and design routes in their development.

Top-down design and software tools

Top-down design refers to design that progressively simplifies a task by breaking it down into several distinct subtasks. At the beginning of the design, the basic requirements for a digital telecommunications system are formulated, for example, the probability of erroneous reception, frequency band, modulation type, etc. At this point, the structure of the telecommunications system as a whole and the implementation features are still unknown. For example, it is not known what part of signal processing will be implemented in the form of hardware, and what − in the form of software, it is not known what parameters low-noise amplifier should have, etc.

Therefore, initially the system is described at a high level, followed by step-by-step division into separate components with a more detailed description. This process continues until the granularity is enough to implement the system at the level of circuit schematic and integrated circuit layout, or until the granularity is sufficient for coding. This approach allows you to split the task into separate subtasks, the solution of which will ultimately comply with the expected result at the beginning of the system design.

Therefore, for example, a block diagram of the linear path of the receiver can be represented as a set of typical functional units − amplifiers, frequency converters, automatic gain control (AGC) devices, etc. Each typical unit can be represented by connecting several operational links − amplifier stage, frequency filter, etc.

Four levels of design are distinguished depending on the level of detail of the task [12]:
− System Level,
− Block Level,
− Circuit/Transistor Level,
− Layout Level.

Fig. 2 illustrates a flow chart of the top-down design and bottom-up verification process. At the design stage at the system level, based on the existing specification, the structure of the telecommunication system is developed, which makes it possible to implement the given specification. In this case, the description of the components of the system is carried out with a high level of abstraction, i.e. without considering the implementation features. For example, a transmitting-receiving device includes a low-noise amplifier, which at the design stage at the system level can be described by the mathematical operation of multiplying a signal by a constant.

After the structure of the system is defined, the design moves to the block diagram level. At this stage, the system is divided into an analog part and a digital part. The digital part consists of the digital signal processing algorithms that will be implemented at the hardware level and at the software level.

Further, the analog and digital parts of the system are also divided into component parts, sub-blocks. The description of the constituent parts is carried out already at the functional level, considering the implementation features. For example, a low-noise amplifier at this design level will be described by
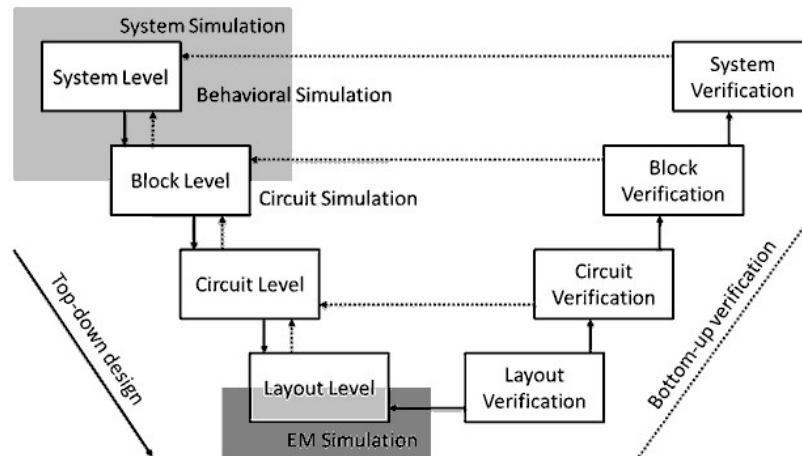
Fig. 2. Top-down design flow chart

a functional model that considers harmonic distortion and amplifier noise. However, the behavioral model will only describe the behavior of the amplifier and will not consider its implementation at the circuit level.

After the structure of the system and the parameters of the system blocks have been determined, design begins at the level of circuit schematic. Analog circuits are described as a netlist of IC elements, and digital circuits are described as a netlist of logical elements. For example, as a result of designing an integrated low-noise amplifier, the circuit schematic of the amplifier and the parameters of the circuit elements, the geometric dimensions of transistors, resistors, capacitors, etc. will be determined.

After the circuit schematics are developed, the design phase begins at the level of the chip layout. At this stage, the integrated circuit is represented by a set of polygons that characterize the layered structure of the integrated circuit. For example, as a result of designing an integrated low-noise amplifier, the arrangement of the circuit elements will be determined, and the conductors connecting the circuit elements will be drawn.

At any stage of the design, it may turn out that the requirements for the parameters of the device being developed cannot be met. In this case, it is necessary to return to a higher level of design and formulate realizable requirements.

The development of transceivers at the modern level involves computer-aided design using simulation of both individual blocks of the system and the entire system. Computer-aided design is made possible by simulating the characteristics of a device or system being developed. Since each design level assumes its own level of abstraction when describing the device being developed, the modeling principles should be different. There are several types of modeling tools:

– system modeling simulator;
– mixed-mode signal simulator;
– circuit simulator;
– electromagnetic simulator and extraction of parasitic parameters of the crystal layout.

System modeling tools are typically used for system-level modeling. This is based on methods of digital signal processing at the software level. These modeling tools differ, as a rule, in a large set of libraries with models of communication channels of transceivers and models of their sub-blocks, functional units and operational links.

Mixed-mode signal simulator are based on an approach that simulates the operation of complex digital circuits and digital signal processing implemented at the hardware level.

Circuit simulator is used to simulate circuits at the transistor level, which allows you to obtain the most accurate characteristics of the device in development. But due to the complexity of the mathematical apparatus, the application of this approach to large circuits is limited.

Electromagnetic simulator and parasitic extraction tools enable the simulation of high frequency passive devices and IC interconnects.

However, the scope of the modeling tool can cover not one, but several levels of design due to the expansion of their functionality. For example, mixed-mode signal simulators can be used for system-level simulations, but the simulations are less convenient. The number of libraries with models of communication channels, transceivers and their components in digital-analog modeling environments is very limited today.

It is advisable to carry out all stages of IC development in one software environment, which makes it possible to implement the principle of end-to-end design. It implies organizing work on a project, transferring the results of one design stage to the next and reflecting changes in all parts of the project. Therefore, software developers strive to cover all stages of development in one software environment.

Historically, Advanced Design System (ADS) produced by Keysight EEsof EDA and AWR Design Environment produced by AWR Corporation are dedicated to MMIC and RF IC design, so the focus has been on frequency domain circuit simulation and electromagnetic analysis. To model the whole system, system simulation tools are used with functional models of analog devices and the launch of circuit simulation when using system modeling.

The Virtuoso Design Environment produced by Cadence Design Systems and Tanner produced by Mentor Graphics were originally focused on the development of digital, mixed signal, and analog ICs using silicon-based technology. The main modeling tools here are mixed-mode signal simulation and circuit analysis of ICs in the time domain. To analyze the parasitic effects of the IC layout, the method of extracting parasitic parameters is used, which allows taking into account the influence of the parasitic effects of the IC layout in circuits with a large number of active elements at relatively low frequencies. For system-level design, mixed-mode modeling tools with a limited set of libraries are available.

With the development of integrated electronic technologies and the integration of all elements of transceiver on a single chip, it becomes necessary to use a wider set of modeling tools, which forces developers to create software that can interact with competitors' products. For example, Keysight Technologies releases the Goldengate software product, which integrates into the Virtuoso design environment from Cadence Design Systems and provides a development environment in which all functions are implemented.

After the layout of the chip has been developed, it is necessary that the system will meet the requirements. For this, bottom-up verification is used (Fig. 2). As part of the bottom-up verification, during a simulation, the parameters of the developed device are determined, and then the obtained parameters are used at the next higher stage of verification. For example, after electromagnetic analysis and during a simulation at the level of the circuit schematic of the low noise amplifier, the decoupling parameter between the output and the input can be determined. This parameter can be substituted into the functional model of the amplifier at the block level simulation. If the obtained result does not meet the specification, it is necessary to repeat the design.

Teaching method in compliance with the top-down design methodology

At Peter the Great St. Petersburg Polytechnic University (SPbPU), within the framework of the international educational master's degree programs "Microelectronics of Telecommunication Systems", students are trained in the transceiver design in accordance with the principle of the top-down design. In a number of courses, students study the principles of transceiver design methods for digital communication systems. Students begin to study the principles of constructing transceiver at the system level, then come to the block level. After that, the construction of blocks of transceivers at the transistor level is studied.

The training begins with an examination of the principles of constructing modulators and demodulators, block diagrams of receivers and transmitters. During the training, students have the opportunity to explore signal processing at various stages and carry out simulations using system simulation tools. So, when studying the design principles of transceivers at the system level, students study the principles of pulse code modulation and passband modulation and detection, block diagrams of modulators and demodulators, block diagrams of linear paths of transceivers, and have the ability to simulate their work using EDA Advanced Design System. At this stage, the functioning of both digital and analog and high-frequency parts of the transceiver is considered. Therefore, for example, in Fig. 3 shows a block diagram of the transceiver with NRZ pulse modulation, taken from a student's laboratory work, and an eye diagram of the
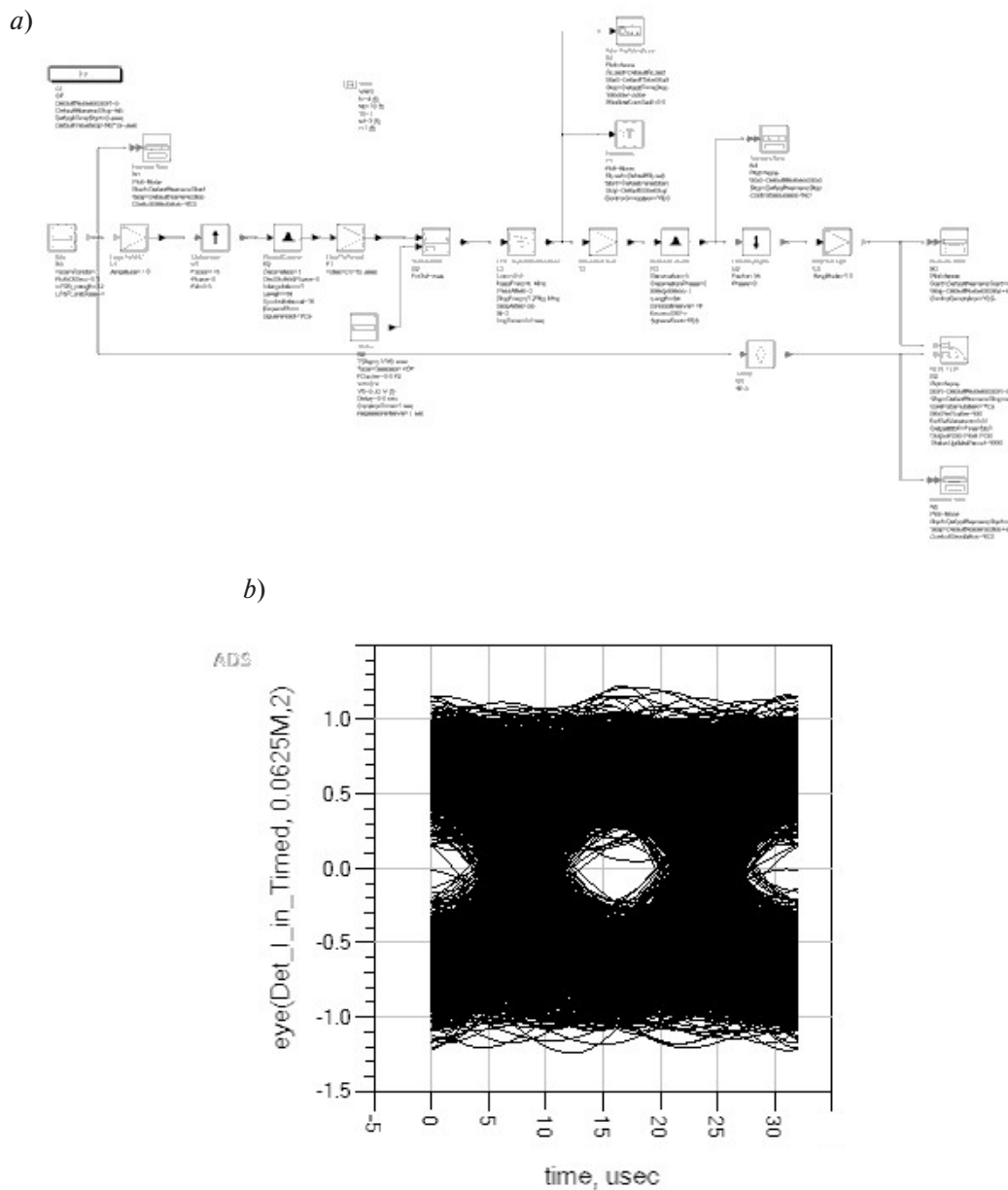
Fig. 3. Block diagram (*a*) and eye diagram (*b*) of a communication system with NRZ pulse modulation
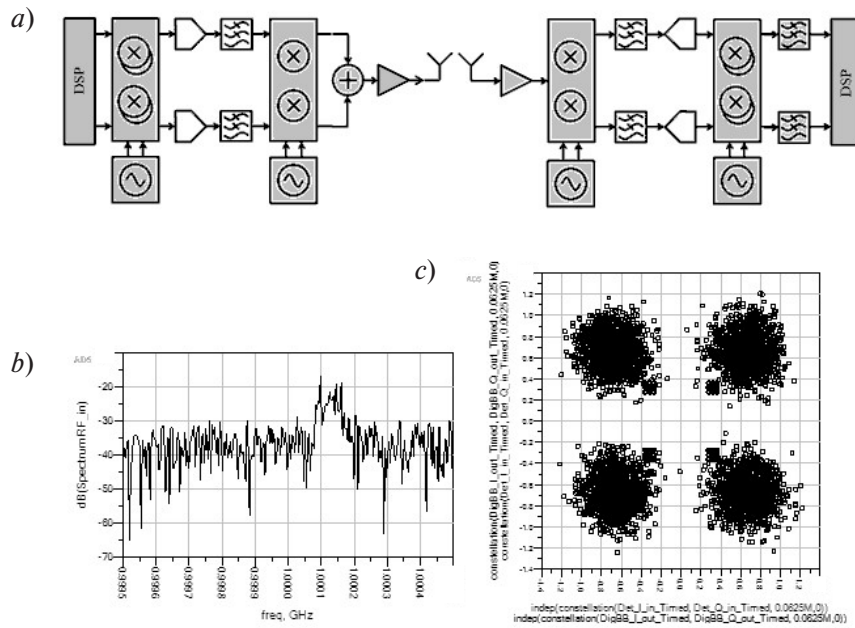
Fig. 4. Block diagram (*a*), spectrum of the signal at the input of the receiver (*b*)
and the signal constellation (*c*) of the communication system with 4-QAM passband modulation
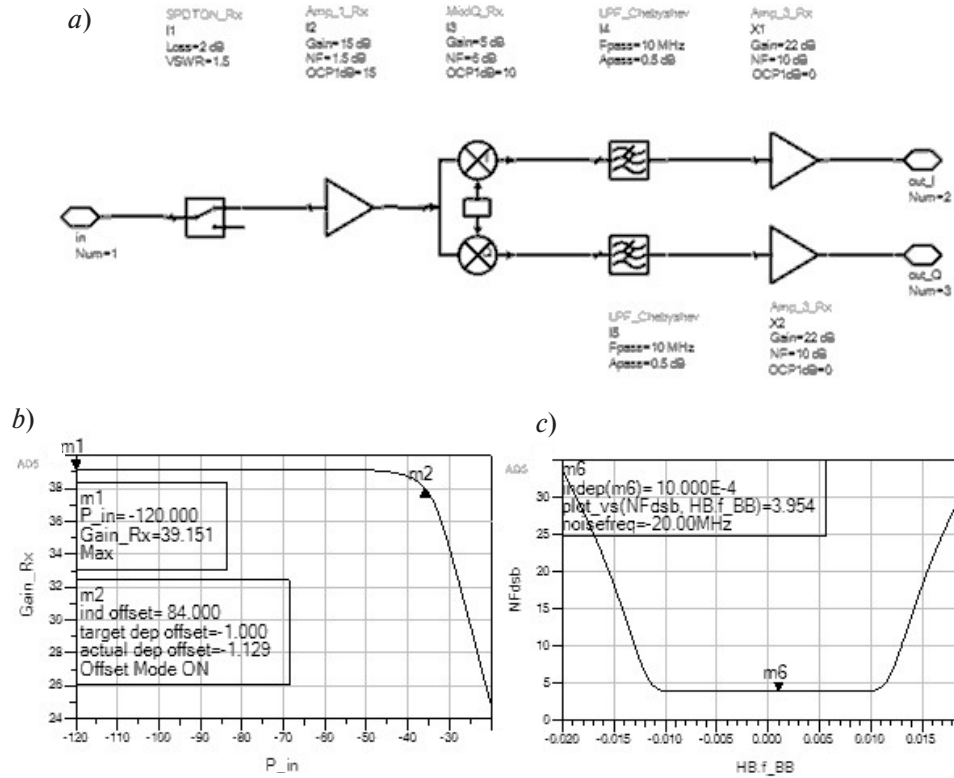
Fig. 5. Block diagram of the receiver (*a*), gain (*b*) and noise figure (*c*) of the receiver

31
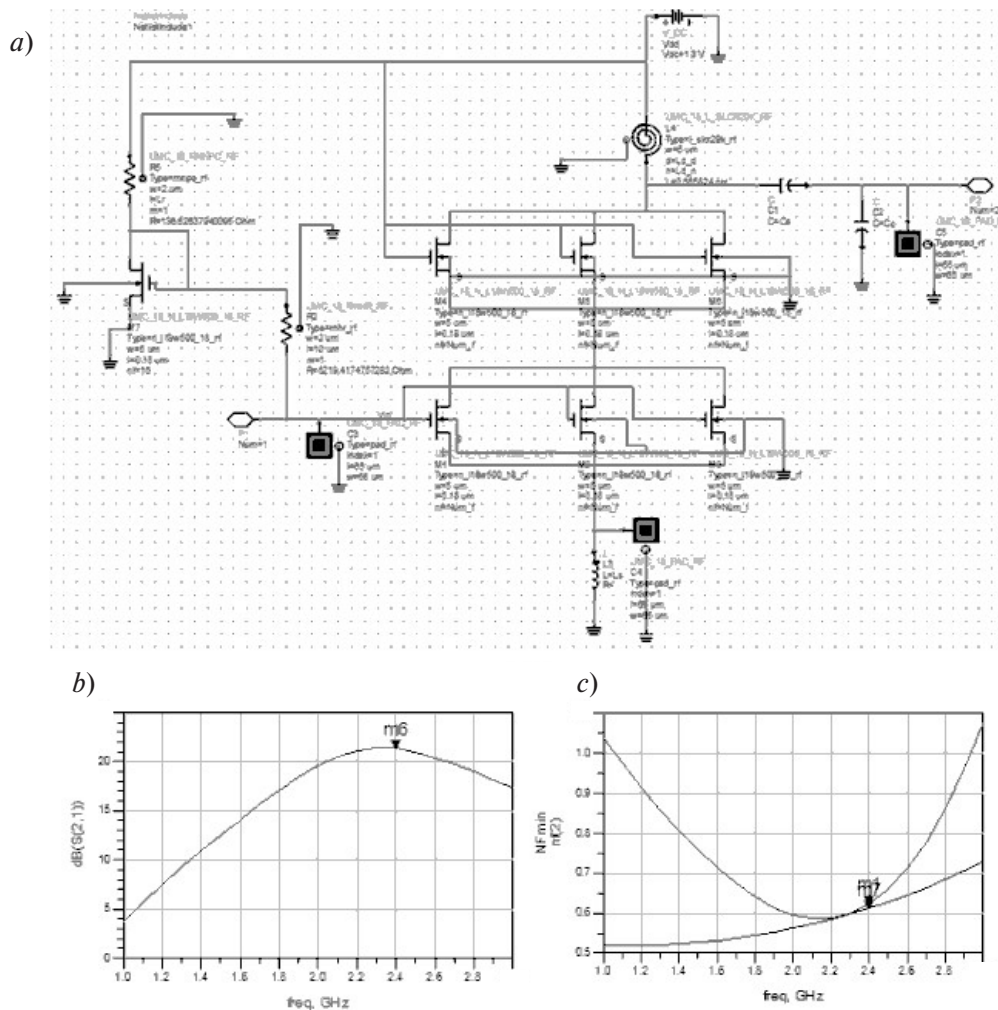
Fig. 6. Schematic diagram of a low-noise amplifier (*a*) and simulation results, gain (*b*) and noise figure (*c*)

signal at the input of the receiver. A block diagram of the transceiver with 4-QAM pulse modulation and the results of modeling the signal spectrum at the receiver input and the signal constellation at the detector input is shown in Fig. 4. In the course of laboratory work, students have the opportunity to investigate signals at various nodes of the circuit and obtain the dependence of the bit error probability on the noise power in the communication channel.

After studying system-level design, students move on to design at the block level. At this stage, students study electrical functional models and parameters of physically realizable units of transceivers, which take into account their noise and nonlinear properties. Students study the influence of the physically realizable block parameters on the characteristics of the entire system. So, for example, Fig. 5 shows a block diagram of the receiving device taken from a student's laboratory work, during the simulation of which the student is introduced to such parameters as the noise figure, the 1 dB compression point.

Moving on to the circuit level, students start the implementation of the main units (amplifiers, mixers, reference oscillators) at the transistor level using PDK of manufacturing integrated circuits. So, for example, one of the student's tasks is to design a low-noise amplifier and simulate its main parameters (Fig. 6) using circuit simulation.

Such an approach in teaching students allows not only to create a holistic view of the subject and basic principles of functioning of transceivers, but also to introduce them to both the top-down design method and the design environment itself.

Conclusion

At the current stage of development of digital communication systems, transmitting and receiving devices are a combination of hardware and software. In their development, it is necessary to use the principles of top-down design and modern design tools. An engineer participating in the development of a transceiver must not only be able to solve his highly specialized task, but also understand the principles of the operation of the system as a whole in order to successfully interact with colleagues within the framework of a large project. In this regard, the use of the methodology for teaching students in accordance with the principles of the top-down design seems appropriate, since this allows not only to introduce the principles of design but also to use development tools. Advanced Design System produced by Keysight Technologies is a good choice of a software tool to study the transceiver top-down design. The EDA ADS provides an opportunity to cover whole design stages from system level design to layout level design and includes system, circuit and electromagnetic simulator to the digital signal processing algorithms, behavioral and circuit modeling. Thus, the students could study the end to end design, that gives them perceptual unit of such a complicated topic as the top-down design of transmitter and receiver.

Acknowledgment

REFERENCES

1. **Proakis J.G., Salehi M.** *Communication systems engineering*. Prentice-Hall, 2002.

2. **Sklar B.** *Digital communications: Fundamentals and applications*. Prentice-Hall, 2001.

3. **Korotkov A.S.** *Ustroystva priyema i obrabotki signalov: mikroelektronnyye vysokochastotnyye ustroystva radiopriyemnikov sistem svyazi* [*Devices for receiving and processing signals: microelectronic high-frequency devices for radio receivers of communication systems*]. St. Petersburg: Polytechnic University Publ., 2010. (rus)

4. **Rogers I.W.M., Plett C., Marsland I.** Radio frequency system architecture design. Artech House, 2013.

5. **Razavi B.** *RF Microelectronics*. Prentice Hall PTR, 1998.

6. **De Los Santos H.J., Sturm Ch., Pontes J.** *Radio systems engineering: A tutorial approach*. Springer, 2015.

7. **Gu Q.** *RF System design of transceivers for wireless communications*. Springer, 2005.

8. **Egan W.F.** *Practical RF system design*. Wiley, Hoboken, 2003.

9. **Baudin P.** *Wireless transceiver architecture: Bridging RF and digital communications*. John Wiley & Sons Ltd, 2015.

10. **Kalivas G.** *Digital radio system design*. Wiley. 2009.

11. **Koivo H., Elmusrati M.** *Systems engineering in wireless communications*. Wiley, 2009.

12. **Frevert R., Haase J., Jancke R., Knochel U., Schwarz P., Kakerow R., Darianian M.** *Modeling and simulation for RF system design*. Springer, 2005.

Received 30.04.2020.

СПИСОК ЛИТЕРАТУРЫ

1. **Proakis J.G., Salehi M.** Communication systems engineering. Prentice-Hall, 2002.

2. **Sklar B.** Digital communications: Fundamentals and applications. Prentice-Hall, 2001.

3. **Коротков А.С.** Устройства приема и обработки сигналов: микроэлектронные высокочастотные устройства радиоприемников систем связи. СПб.: Изд-во Политехн. ун-та, 2010.

4. **Rogers I.W.M., Plett C., Marsland I.** Radio frequency system architecture design. Artech House, 2013.

5. **Razavi B.** RF microelectronics. Prentice Hall PTR, 1998.

6. **De Los Santos H.J., Sturm Ch., Pontes J.** Radio systems engineering: A tutorial approach. Springer, 2015.

7. **Gu Q.** RF system design of transceivers for wireless communications. Springer, 2005.

8. **Egan W.F.** Practical RF system design. Wiley, Hoboken, 2003.

9. **Baudin P.** Wireless transceiver architecture: Bridging RF and digital communications. John Wiley & Sons Ltd, 2015.

10. **Kalivas G.** Digital radio system design. Wiley, 2009.

11. **Koivo H., Elmusrati M.** Systems engineering in wireless communications. Wiley, 2009.

12. **Frevert R., Haase J., Jancke R., Knochel U., Schwarz P., Kakerow R., Darianian M.** Modeling and simulation for RF system design. Springer, 2005.

Статья поступила в редакцию 30.04.2020.

THE AUTHOR / СВЕДЕНИЯ ОБ АВТОРЕ

Balashov Evgenii V.
Балашов Евгений Владимирович
E-mail: balashov_ev@mail.ru

Software of Computer, Telecommunications and Control Systems

AUTOMATIC GENERATION OF SOFTWARE BUG FIXES BASED ON ANALYSIS OF SOFTWARE REPOSITORIES

A. Belskii, V.M. Itsykson

Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation

This paper describes the method, which is developed by the authors to automated correction of software errors, which is based on the analysis of successful project fixes for the ABAP programming language available in open repositories. The method generates the candidates of patches based on predefined templates and ranks the results by the probability of successful application, which is determined by a probabilistic model using machine learning methods. The probabilistic model is formed by training on features, which are extracted from data from successful and unsuccessful patches of ABAP programs in open repositories. The developed method is tested on synthetic examples and real projects with errors in the ABAP language. As a result of the experiments, the method successfully generated some patches, which showed their efficiency. The results in accuracy and efficiency are comparable or superior to the results of experiments in similar works by other authors.

АВТОМАТИЧЕСКОЕ ФОРМИРОВАНИЕ ИСПРАВЛЕНИЙ ОШИБОК ПРОГРАММНОГО КОДА НА ОСНОВЕ АНАЛИЗА ПРОГРАММНЫХ РЕПОЗИТОРИЕВ

А. Бельский, В.М. Ицыксон

Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация

Описан разработанный подход к автоматизированному исправлению программных ошибок на основе анализа успешных исправлений проектов для языка программирования ABAP, имеющихся в открытых репозиториях. Подход основан на генерации кандидатов на исправления (патчей) по заранее определенным шаблонам и ранжирует полученные результаты по вероятности успешного применения, определяемой на основании вероятностной модели, полученной с помощью методов машинного обучения. Вероятностная модель формируется за счет обучения на свойствах, извлекаемых из данных успешных и неуспешных патчей ABAP-программ, доступных в открытых репозиториях. Разработанный подход протестирован как на искусственных примерах, так и на реальных проектах на языке ABAP с ошибками. В результате проведенных экспериментов успешно сформирован ряд патчей, которые показали свою работоспособность. Результаты по точности и эффективности сопоставимы или превосходят результаты экспериментов в аналогичных работах других авторов.

Introduction

In recent years the size of software has been constantly growing and the development cycle is shortening, which usually leads to a total decrease in the quality of software products. This is unacceptable in areas such as embedded systems in medicine, energy, engineering, the financial sector, and others, or even it can lead to significant material losses or danger to human life and health.

To overcome these problems, developers use various methods to improve the quality of software, like testing, verification, or the static analysis of software. However, all common methods of improving the quality of software have certain limitations and they cannot fully guarantee the quality of programs. For example, testing may detect errors in software, but it cannot guarantee that there are no errors in the tested software. In addition, there are entire classes of programs, such as parallel systems, whose behavior may be non-deterministic and whose testing is inefficient. Formal methods, such as deductive verification and static analysis, are still limited by the size of the analyzed programs and can only be applied to a narrow area of software projects.

Recently, software engineering has actively been conducting research in the field of analysis and application of the accumulated experience of millions of programmers in writing hundreds of thousands of software projects. This experience is recorded in software repositories (Version control systems, VCS) as a history of changes to projects and comments to commits, as well as in task and error management systems (issue tracking and bag tracking) as a history of changes to tasks and errors. There are a large number of methods that analyze the accumulated information and extract it from the knowledge, which is used in solving various problems of software engineering. These methods have proven themselves well in various areas of software engineering. Recently, these approaches have been applied in the field of detecting and correcting software errors, using not only the artifacts of analyzed software projects, but also the previously untapped potential of information, which is stored in hundreds of thousands of software repositories and allowing to reuse and generalize the experience of millions of developers.

This paper describes the results of the research in the field of automated error correction of software code based on the experience of analyzing successful fixes (patches) of many projects for the ABAP programming language [1], which is widely used in SAP software products.

The article is organized as follows: The first section contains a description of the task and a brief overview of the subject area. The second section illustrates the scheme and verbal description of the method, which is developed by the authors. The third section is devoted to the detailed description of the developed method and includes algorithms, a mathematical model, and the technologies and used methods. The fourth section shows the results of testing of the developed method and the analysis of the results. In conclusion, the results are summarized, the results are evaluated and plans for further research are formulated.

Related work

Nowadays, there are various technologies that allow for the automatic generation of bug fixes in programs (patches). The following methods can be the most representative of these technologies.

GenProg [2], relifix [3], Astor [4], and history-based program repair [5] methods, which are based on genetic programming technology. This class of methods is a method of stochastic problem solving, which

is based on the ideas of evolutionary genetics, which include the genotype (the genetic material of an individual) stored in memory, differential reproduction of these genotypes, and variations, which are created by processes, which are similar to the biological processes of mutation and crossover [6].

Methods SemFix [7], JFIX [8], CRSearcher [9], Qlose [10], Semantic program repair using a reference implementation [11], Static automated program repair for heap properties [12], Automated program repair with canonical constraints [13], which are based on the semantic approach. The main idea of these methods is to define a set of restrictions for an expression with an error by applying methods of symbolic program execution [14] and solving these restrictions by using various SMT solvers [15].

Methods R2Fix [16], Prophet [17], ELIXIR [18], Data-Guided Repair of Selection Statements [19], which are based on a class of machine learning methods [20]. The main idea of these methods is to build machine learning models [21], which is based on the source code of programs with errors and their corrections, as well as comments and other data from source code repositories such as GitHub and others. Then the trained model is used for classification tasks, for example, to solve problems of detecting errors in the source code of the program or determining suitable patches that are classified on the same parameters as the error.

The main disadvantage of methods, which are based on genetic programming technology, is the random selection of all possible patch variants without analyzing both the source code context with an error and similar patches. Also, methods, which are based on the semantic approach, already widely analyze the source code context with an error, but do not use the experience of similar patches to strengthen the algorithm for automatic patch generation. At the same time, methods, which are based on the class of machine learning methods, are the closest in implementation to the given task for the authors, since they analyze both the source code context with an error and similar patches.

Thus, the goal of this research is to develop a method to automatically generate bug fixes for software code based on the previously accumulated experience of creating patches. The method has to have an algorithm, which is based on machine learning methods and allows for the automatic generation of patches for various types of errors of the program code without using specifications and other means of automated code generation.

Overview

The main idea of the proposed method is to automatically generate patches for errors in ABAP programs by generating candidate patches based on predefined templates and ranking the results by the probability of successful application, which is determined based on a probabilistic model, which is obtained by using machine learning methods. In turn, the probabilistic model is formed by learning from the data of successful and unsuccessful patches of ABAP programs. The main idea of the method is presented in Fig. 1.

The method contains two main contours − "Machine learning model training" and "Patch generation and ranking", which contain the following seven functional blocks.

Block 1 "Forming an abstract syntax tree". The abstract syntax tree (AST) [22] is based on the source code with an error and a patch. Two independent AST are formed by applying the recursive descent method [23] based on the text of the source code of the ABAP program containing the error and the correction of this error (patch). Further work on the analysis of the source code of the ABAP program is performed on the AST, which gives a more accurate data of the types of elements of the ABAP program (variables, constants, operators, etc.) and their relationships.

Block 2 "Determination of the features of a successful patch". The features of successful patches were formulated to train the probabilistic model, which are determined by analyzing the AST of the source code with an error and the AST of the source code of the patch, which is obtained in block 1. For example, if the program correction was formed by adding a check for an empty variable value before executing the division operator, this feature can be used as a feature of the successful patch and used for training the probabilistic model.
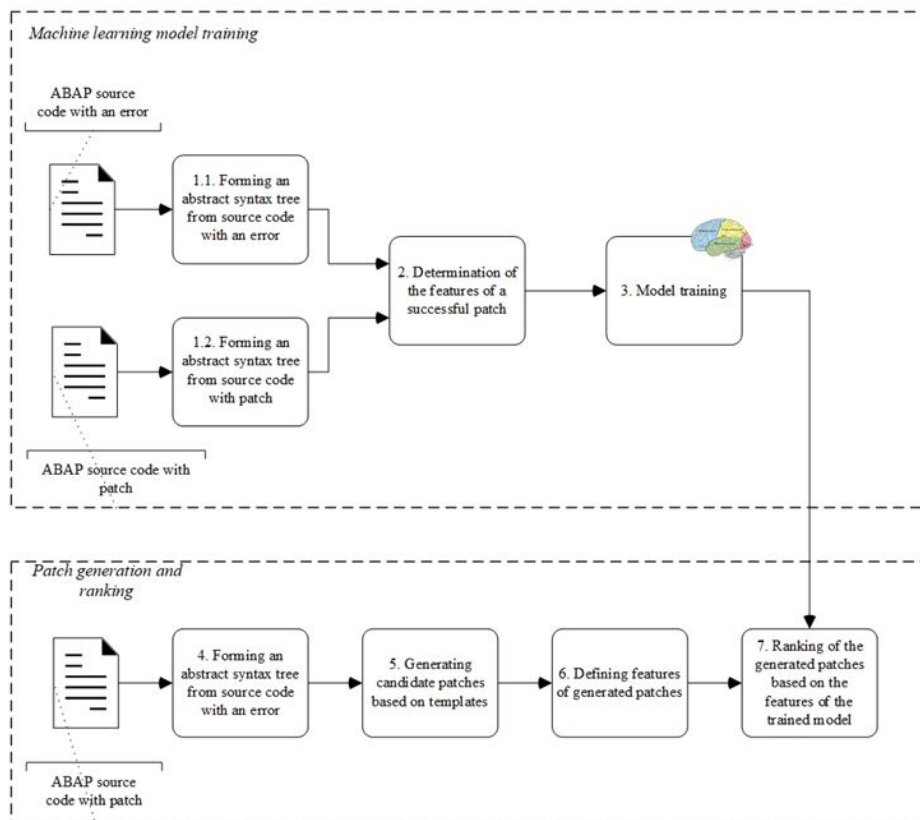
Fig. 1. Scheme of the proposed method

Block 3 "Model training". In this block, the machine learning model is trained based on the features of successful patches, which are obtained in block 2. As a result, the trained model can be used to predict the success rate of any ABAP patch.

Block 4 "Forming an abstract syntax tree from source code with an error". The source code of the ABAP program that needs to be automatically generated for a patch is used to generate the AST, similar to block 1.

Block 5 "Generating candidate patches based on templates". Data of all variables and constants based on the AST is extracted from the program with an error. Furthermore, the array of possible conditions is generated from the data of all variables and constants. Finally, patches are generated using templates based on the received array of variables/constants and the array of possible conditions.

Block 6 "Defining features of generated patches". The features of generated patches in block 5 are determined in the same way as successful patches in block 2 are.

Block 7 "Ranking of the generated patches based on the features of the trained model". The probability of a successful patch is determined for each generated patch based on the trained model, which is obtained in block 3 and the features of the generated patches, which are obtained in block 6. The resulting list of generated patches is sorted in descending order of the probability of a successful patch. Generated patches with the highest probability of successful patches are considered target patches.

Our approach

Let's look more detailed at the stages of the method and the nuances of implementing the methods and models, which are shown in Fig. 1.

Generating AST from source code

The program must be translated into a formalized view to perform the analysis that is suitable for further processing. In this paper, we use an abstract syntax tree. Since there is no official grammar for parser

generators (for example, for ANTLR) for the ABAP language, the authors developed a lightweight parser based on the recursive descent method. The actual formation of the AST from the source code is performed in blocks 1 and 4, which are shown in Fig. 1. The goal of it is to more accurately determine the types of objects and their relationships for further analysis of ABAP programs. The simplified algorithm for parsing ABAP programs is presented as pseudocode in Listing 1.

In line 1 of the algorithm, the input data is the array of $str \in S$, which is the source code lines of the ABAP program. In lines 2-8, the array of lexemes L is generated for each string of str in the source code. In lines 9-13, the array of tokens $t \in T$ is formed by defining the following data for each lexem l from the array of lexemes L:

— the token type t_t (header, operator, brackets, number, variable, type), which is defined by assigning each token to a programming language object class;

— the error flag of the error token b_t, is determined by fulfilling the condition: if the source code line str contained an error, then all tokens t, which were related to tokens L, will have the value true.

— In lines 14-20, the array of nodes AST P is formed from the token array T. Each node $p \in P$ is the following:

$$\langle p_p | l | t_n | b \rangle \in P,$$

where p_p — a reference to the parent node $p \in P$; l — a lexeme; t_n — a node type, which is determined from the token type t_t; b — the error flag of the node is determined from the error flag of the error token b_t.

The array of nodes AST P is formed using the recursive descent method, which consists of recursively traversing the entire array of tokens $t \in T$ and building their relationships through references p_p according to the grammatical rules of the programming language ABAP, which is shown in Fig. 2.

```
1 Input: S
2 for str in S do {
3       for element in str do {
4               L(l) = element
5       }
6 }
7 for l in L do {
8       t(l) = l
9       t(t_t) = defClass(l)
10      t(b_t) = defBug(str)
11 }
12 for <l, t_t, b_t> in T do {
13      P(p_p) = defParent(P)
14      P(l) = l
15      P(t_n) = t_t
16      P(b) = b_t
17 }
```

Listing 1. AST generation algorithm
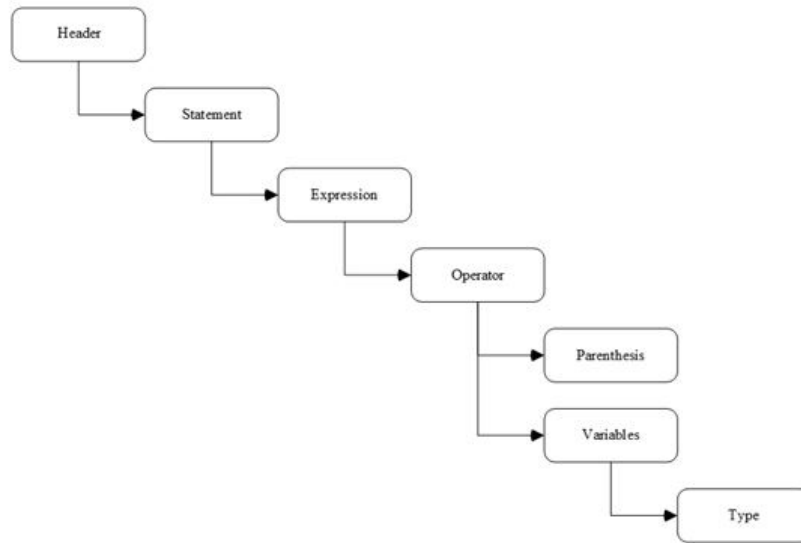
Fig. 2. Grammatical rules of the programming language ABAP

Defining patch features

Patch features are defined in blocks 2 and 6, which is shown in Fig. 1. The authors of the method formulated 15 patch features based on the results of many years of experience working with the ABAP programming language on real projects and creating thousands of bug fixes, as well as on the results of analyzing patches to ABAP programs from open source repositories, which are shown in Table 1. These features are extracted from the source code of ABAP programs with an error and a patch. Previously, to determine the features the method defines the differences between P_{bug} AST source code with an error and P_{patch} AST source code with a patch in the form of node indexes of the beginning of the difference $idx_{start}(P_{patch})$ and the end of the difference $idx_{end}(P_{patch})$. Also, a list of all patch variables $v \in V(P_{patch})$ is defined within $idx_{start}(P_{patch})$ and $idx_{end}(P_{patch})$.

Model training

Model training is performed in block 3 in Fig. 1. There are a number of models with their own advantages and disadvantages to solve classification problems with a teacher in machine learning. The authors of the method chose the logistic regression model [24], because with a small number of properties, this model shows better performance with similar accuracy than other machine learning methods, such as neural networks or the support vector machine. Moreover, the logistic regression model is more convenient to implement and adapt [25], and is also widely used in similar works by other authors.

The following matrix $m \times 15$ is used to train the model:

F_{11}	F_{12}	F_{13}	F_{14}	F_{15}	F_{16}	F_{17}	F_{18}	F_{19}	F_{110}	F_{111}	F_{112}	F_{113}	F_{114}	F_{115}
F_{21}	F_{22}	F_{23}	F_{24}	F_{25}	F_{26}	F_{27}	F_{28}	F_{29}	F_{210}	F_{211}	F_{212}	F_{213}	F_{214}	F_{215}
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
F_{m1}	F_{m2}	F_{m3}	F_{m4}	F_{m5}	F_{m6}	F_{m7}	F_{m8}	F_{m9}	F_{m10}	F_{m11}	F_{m12}	F_{m13}	F_{m14}	F_{m15}

where m — number of examples in the form of features P_{bug} and P_{patch} (see the section Defining patch features).

Patch features

Feature name	Algorithm for determining
Type of error F_1	Determined manually, possible options: 1 — division by 0. 2 — using an empty pointer. 3 — error in the conditional operator. 4 — error in the loop condition
Type of modification the patch F_2	1. Adding a check. If the nodes in the tree P_{patch} within $idx_{start}(P_{patch})$ and $idx_{end}(P_{patch})$, where $\exists l = $ if, то $F_2 = 1$. 2. The change of the if condition. If the nodes in the tree P_{patch} within $idx_{start}(P_{patch})$ and $idx_{end}(P_{patch})$, where $\forall l \neq $ if, but the tree nodes associated with them are $p_p \in P_{patch}$, where $\exists l = $ if, then $F_2 = 2$. 3. The change of the loop condition. If the nodes in the tree P_{patch} within $idx_{start}(P_{patch})$ and $idx_{end}(P_{patch})$, where $\forall l \neq $ loop, but the tree nodes associated with them are $p_p \in P_{patch}$, where $\exists l = $ loop, then $F_2 = 3$. 4. Otherwise $F_2 = 4$
Location of the patch modification F_3	The tree nodes with error P_{bug} are defined by defining $idx_{start}(P_{bug})$ and $idx_{end}(P_{bug})$ of the tree nodes P_{bug}, where $\exists b = $ true. Further, the place where the patch modification occurs is determined by the following rule based on the location data $idx_{start}(P_{patch})$ and $idx_{end}(P_{patch})$ of the tree nodes Ppatch and the location data idxstart(Pbug) and idxend(Pbug) of the tree nodes Pbug: 1. If $idx_{start}(P_{bug}) >= idx_{start}(P_{patch})$ and $idx_{end}(P_{bug}) >= idx_{end}(P_{patch})$, then $F_3 = 0$. 2. If $idx_{start}(P_{bug}) < idx_{start}(P_{patch})$ and $idx_{end}(P_{bug}) >= idx_{end}(P_{patch})$, then $F_3 = 1$. 3. Иначе $F_3 = 2$
If operator is present at the error location F_4	If the tree nodes P_{bug}, where $\exists b = $ true and $\exists l = $ if, then $F_4 = 1$ else 0
Loop operator is present at the error location F_5	If the tree nodes P_{bug}, where $\exists b = $ true and $\exists l = $ loop, then $F_5 = 1$ else 0
/,*,+,- operators are present at the error location F_6	If the tree nodes P_{bug}, where $\exists b = $ true and $\exists l = $ /,*,+,-, then $F_6 = 1$ else 0
Call operator is present at the error location F_7	If the tree nodes P_{bug}, where $\exists b = $ true and $\exists l = \Rightarrow$, then $F_7 = 1$ else 0
Variable is present at the if operator at the patch F_8	Defining the tree nodes P_{patch} within $idx_{start}(P_{patch})$ and $idx_{end}(P_{patch})$, where $\exists l = $ if. Further, if in the defined nodes $\exists l = v$, then $F_8 = 1$ else 0
Variable is present at the loop operator at the patch F_9	Defining the tree nodes P_{patch} within $id_{xstart}(P_{patch})$ and $idx_{end}(P_{patch})$, where $\exists l = $ loop. Further, if in the defined nodes $\exists l = v$, then $F_9 = 1$ else 0
Variable is present at the /,*,+,- operators at the patch F_{10}	Defining the tree nodes P_{patch} within $idx_{start}(P_{patch})$ and $idx_{end}(P_{patch})$, where $\exists l = $ /, *,+,-. Further, if in the defined nodes $\exists l = v$, then $F_{10} = 1$ else 0

Variable is present at the call operator at the patch F_{11}	Defining the tree nodes P_{patch} within $idx_{start}(P_{patch})$ and $idx_{end}(P_{patch})$, where $\exists l = \Rightarrow$. Further, if in the defined nodes $\exists l = v$, then $F_{11} = 1$ else 0
Variable is present at the if operator the error location F_{12}	Defining the tree nodes P_{bug}, where $\exists b = true$, and $\exists l = if$. Further, if in the defined nodes $\exists l = v$, then $F_{12} = 1$ else 0
Variable is present at the loop operator the error location F_9	Defining the tree nodes P_{bug}, where $\exists b = true$, and $\exists l = loop$. Further, if in the defined nodes $\exists l = v$, then $F_{13} = 1$ else 0
Variable is present at the /,*,+,- operators the error location F_{10}	Defining the tree nodes P_{bug}, where $\exists b = true$, and $\exists l = /,*,+,-$. Further, if in the defined nodes $\exists l = v$, then $F_{14} = 1$ else 0
Variable is present at the call operator the error location F_{11}	Defining the tree nodes P_{bug}, where $\exists b = true$, and $\exists l = \Rightarrow$. Further, if in the defined nodes $\exists l = v$, then $F_{15} = 1$ else 0

The training is performed for the logistic regression model:

$$prediction = \frac{1}{1 + e^{-\theta \times F}}.$$

The main idea of training a logistic regression model is to determine coefficients θ for features F successful patches (see the section Defining patch features), which can then be used to build a forecast prediction for any generated patches ABAP programs based on their features. The coefficients θ are determined using the gradient descent method [26], according to which the following calculations are performed simultaneously:

$$\theta_0 = \theta_0 - \alpha \times \frac{1}{m}(prediction - y),$$

$$\theta_1 = \theta_1 - \alpha \times \frac{1}{m}(prediction - y) \times F_1 + \frac{\lambda}{m} \times \theta_1,$$

$$...$$

$$\theta_{15} = \theta_{15} - \alpha \times \frac{1}{m}(prediction - y) \times F_{15} + \frac{\lambda}{m} \times \theta_{15},$$

where y – the result of successful application P_{patch} to P_{bug} (it is set manually, 0 – unsuccessful patch, 1 – successful patch); α – the coefficient of speed of learning (it is set manually and is used to regulate the accuracy and speed of the determination process θ); λ – the regularization coefficient (it is set manually and used to reduce the likelihood of model overfitting).

When calculating the coefficients θ the cost function J is also calculated, which should tend to zero at each iteration of the calculation and reflects the progress and correctness of the gradient descent method:

$$J = \frac{1}{m} \times \sum_{i=1}^{m}\left(-y_i \times \log(prediction) - (1 - y_i) \times \log(1 - prediction)\right) + \frac{\lambda}{2 \times m} \times \sum_{j=2}^{15} \theta_j^2.$$

```
1 Input: P_fix
2 v_fix = defVarible(P_fix)
3 for v_fix1 in V_fix do {
4      for v_fix2 in V_fix do {
5              CND_fix(cnd_fix) = v_fix1 > v_fix2
6              CND_fix(cnd_fix) = v_fix1 < v_fix2
7              CND_fix(cnd_fix) = v_fix1 = v_fix2
8              CND_fix(cnd_fix) = v_fix1 ≠ v_fix2
9      }
10     CND_fix(cnd_fix) = v_fix1 is initial
11     CND_fix(cnd_fix) = v_fix1 is not initial
12 }
13 for cnd_fix in CND_fix do {
14     GeneratePatchAddIf (cnd_fix)
15     GeneratePatchEditIf (cnd_fix)
16     GeneratePatchEditCycle (cnd_fix)
17 }
```

Listing 2. Algorithm for generating candidate patches based on templates

Generating candidate patches based on templates

The generation of patch candidates by templates is performed in block 5 in the method diagram in Fig. 1. The generation of patch candidates by predefined templates is performed from the source code objects of the ABAP program with an error. This algorithm is presented in Listing 2.

Line 2 defines the array of variables V_{fix} of the tree nodes P_{fix}, which was obtained by forming AST (see the section Generating AST from source code) from the text of the program to automatically generate the patch for. The array of variables V_{fix} is determined from l of the tree nodes $p_{fix} \in P_{fix}$, where $\exists b = $ true and $\exists t_n = $ Variable. In lines 3-14, the array of conditions CND_{fix} is generated by executing the Cartesian product of the array of variables V_{fix} and the array of degrees of comparison ($>$, $<$, $=$, $\neq$, is initial, is not initial). In lines 15-17, patch candidates $P_{fixpatch}$ are generated by adding a check (if statement) with the condition cnd_{fix} from the array of conditions CND_{fix} before the error location. In lines 15-18, patch candidates $P_{fixpatch}$ are generated by replacing a condition in the check statement (if) with cnd_{fix} from the array of conditions CND_{fix} in the error location. In lines 15-19, patch candidates $P_{fixpatch}$ are generated by changing the condition in the loop operator to cnd_{fix} from the array of conditions CND_{fix} in the error location.

Further, the features for the generated patch candidates $P_{fixpatch}$ are defined (see the section Defining patch features) and the application success rate is determined (see the section Ranking generated patches based on features and the trained logistic regression model).

Ranking generated patches based on features and the trained logistic regression model

The ranking of generated patches based on features and the trained logistic regression model is performed in block 7 in Fig. 1. The success rate $prediction_{fixpatch}$ is determined for each generated patch candidate $P_{fixpatch}$ (see the section Generating candidate patches based on templates) by applying the trained logistic regression model:

$$prediction_{fixpatch} = \frac{1}{1 + e^{-\theta \times F_{fixpatch}}},$$

where θ obtained in the process of training the logistic regression model (see the section Model training); $F_{fixpatch}$ obtained in the process of defining patch features for patch candidates $P_{fixpatch}$ (see the section Defining patch features).

Further, $P_{fixpatch}$ with the maximum value of $prediction_{fixpatch}$ is selected, which means that the candidate patches with the highest probability of application success are selected based on the analysis of existing patches.

Evaluation

The method was tested on 10 projects in the ABAP language with errors. Some of the examples were prepared by the authors in accordance with the required types of errors for evaluating the method's performance, while the other part − real projects. The test results are shown in table 2.

T a b l e 2

The results of the test method

Name of the source code example	Type of error	Number of lines of source code	Number of candidate patches generated	Execution time, sec	The patch was successfully generated
ABAPException.abap[1]	Division by 0	34	300	66	Yes
mycalculator.abap[2]	Division by 0	25	100	14	Yes
SubRoutines.abap[3]	Division by 0	59	1200	836	Yes
AbapRep_usingclassHana.abap[4]	Calling a function using an empty pointer	27	800	371	No
zma_dp_strategy.prog.abap[5]	Calling a function using an empty pointer	33	700	285	Yes
zcl_pi_static.clas.abap[6]	Calling a function using an empty pointer	46	100	12	Yes
TestCodeWithIfBug.abap[7]	Error in the if operator	17	200	37	No
TestCodeWithIfBug2.abap[8]	Error in the if operator	11	50	9	Да
TestCodeWithCycleBug.abap[9]	Error in the loop operator	13	20	8	1

[1] https://github.com/naveenkumarbaskaran/SAP_ABAP19Jan/blob/efc47953337bb8fbaeee506ee9a3c701bfa4f498/ABAPException.abap
[2] https://github.com/naveenkumarbaskaran/SAP_ABAP19Jan/blob/master/mycalculator.abap
[3] https://github.com/naveenkumarbaskaran/SAP_ABAP19Jan/blob/master/SubRoutines.abap
[4] https://github.com/naveenkumarbaskaran/SAP_ABAP19Jan/blob/master/AbapRep_usingclassHana.abap
[5] https://github.com/Huargh/OO-Design-Patterns-in-ABAP/blob/master/src/zma_dp_strategy.prog.abap
[6] https://github.com/ivangurin/abapPI/blob/5f30db0cc7a408a759ad833fe14f6e803b1b46bf/src/zcl_pi_static.clas.abap
[7] https://github.com/AlekseiBelskii/AlexB/blob/master/TestCodeWithIfBug.abap
[8] https://github.com/AlekseiBelskii/AlexB/blob/master/TestCodeWithIfBug2.abap
[9] https://github.com/AlekseiBelskii/AlexB/blob/master/TestCodeWithCycleBug.abap

TestCodeWithCycleBug2.abap[1]	Error in the loop operator	13	50	12	0
		276	3520	1650	6/10

The first column shows the project name with an error and a link to Github source code repository. The second column shows the type of error that patches were generated for. The third column shows the number of lines of source code with an error. The fourth column contains the number of error correction candidate patches generated for each project. The number of candidate patches was formed in an amount, which was enough to get the expected result. The fifth column shows the time it took to generate candidate patches for each project with an error. The last column shows whether patches were successfully generated for each project with an error or not. Patch is considered successfully generated if the desired patch is found among all the generated patch candidates with the highest probability of success $prediction_{fixpatch}$:

The method was tested on a stand with the following characteristics: Intel Core i3-7100U 2.40 Ghz, 4.00 Gb RAM, Windows 10. As a result of the experiments, 6 patches were successfully found for 10 programs with an error of 1650 seconds, which indicates the reality of using machine learning methods for automatic patch generation, but at the same time, the obtained accuracy and the speed indicate the necessity for additional tests, better training of the logistic regression model, increasing the power of the test stand, as well as other improvements to the method. These improvements are expected to be developed and implemented in future works.

Conclusion

During the research, the method was developed to automatically generate bug fixes for ABAP programs based on the analysis of existing patches, which generates candidate patches for ABAP programs and ranks the results using machine learning methods. The obtained preliminary test results suggest that using machine learning methods to solve problems of automatic error correction in programs is a promising direction for software engineering. Directions for further development of the work:
- conducting deeper testing of the method on a wider set of real projects;
- extending the method to support new programming languages;
- extending the set of the extracted features and the list of error types to fix;
- use more complex machine learning models to improve the performance of the method.

REFERENCES

1. SAP SE. ABAP—Keyword Documentation. Available: *https://help.sap.com/doc/abapdocu_latest_index_htm/latest/en-US/index.htm −2019.*

2. **Le Goues C., et al.** Genprog: A generic method for automatic software repair. *IEEE Transactions on Software Engineering*, 2012, Vol. 38, No. 1, P. 54.

3. **Tan S.H., Roychoudhury A.** relifix: Automated repair of software regressions. *Proceedings of the 37th International Conference on Software Engineering*. IEEE Press, 2015, Vol. 1, Pp. 471−482.

4. **Martinez M., Monperrus M.** Astor: A program repair library for java. *Proceedings of the 25th International Symposium on Software Testing and Analysis*. ACM, 2016, Pp. 441−444.

5. **Le X.B.D., Lo D., Le Goues C.** *History driven automated program repair*. 2016.

6. **Forrest S.** Genetic algorithms: Principles of natural selection applied to computation. *Science*, 1993, Vol. 261, Pp. 872−878.

---

[1] https://github.com/AlekseiBelskii/AlexB/blob/master/TestCodeWithCycleBug2.abap

7. **Nguyen H.D.T., et al.** Semfix: Program repair via semantic analysis. *Proceedings of the 35th International Conference on Software Engineering (ICSE)*. IEEE, 2013, Pp. 772−781.

8. **Le X.B.D., et al.** JFIX: semantics-based repair of Java programs via symbolic PathFinder. *Proceedings of the 26th ACM SIGSOFT International Symposium on Software Testing and Analysis*, ACM, 2017, Pp. 376−379.

9. **Wang Y., et al.** CRSearcher: Searching Code Database for Repairing Bugs. *Proceedings of the 9th Asia-Pacific Symposium on Internetware*, ACM, 2017, P. 16.

10. **D'Antoni L., Samanta R., Singh R.** Qlose: Program repair with quantitative objectives. *International Conference on Computer Aided Verification*. Springer, Cham, 2016, Pp. 383−401.

11. **Mechtaev S., et al.** Semantic Program Repair Using a Reference Implementation. *Proceedings of ICSE*, 2018.

12. **van Tonder R., Le Goues C.** *Static Automated Program Repair for Heap Properties*, 2018.

13. **Hill A., Păsăreanu C.S., Stolee K.T.** Automated program repair with canonical constraints. *Proceedings of the 40th International Conference on Software Engineering: Companion Proceeedings*. ACM, 2018, Pp. 339−341.

14. **Cadar C., et al.** KLEE: Unassisted and Automatic Generation of High-Coverage Tests for Complex Systems Programs. *OSDI*, 2008, Vol. 8, Pp. 209−224.

15. **De Moura L., Bjørner N.** Z3: An efficient SMT solver. *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*. Springer, Berlin, Heidelberg, 2008, Pp. 337−340.

16. **Liu C., et al.** R2Fix: Automatically generating bug fixes from bug reports. *Proceedings of the 6th International Conference on Software Testing, Verification and Validation*, IEEE, 2013, Pp. 282−291.

17. **Long F., Rinard M.** Automatic patch generation by learning correct code. *ACM SIGPLAN Notices*, 2016, Vol. 51, No. 1, Pp. 298−312.

18. **Saha R.K., et al.** Elixir: Effective object-oriented program repair. *2017 32nd IEEE/ACM International Conference on Automated Software Engineering (ASE)*, IEEE, 2017, Pp. 648−659.

19. **Gopinath D., et al.** Data-guided repair of selection statements. *Proceedings of the 36th International Conference on Software En***gineering. ACM, 2014, Pp. 243-253.**

20. **Dietterich T.G.** *Machine learning. Encyclopedia of Computer Science*. John Wiley and Sons Ltd., GBR, 2003, Pp. 1056−1059.

21. **Witten I.H., et al.** *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.

22. **Cui B., et al.** Code comparison system based on abstract syntax tree. *Proceedings of the 3rd IEEE International Conference on Broadband Network and Multimedia Technology (IC-BNMT)*. IEEE, 2010, Pp. 668−673.

23. **Matthew S. Davis.** An object oriented approach to constructing recursive descent parsers. *SIGPLAN Not.*, 2000, 35, 2, Pp. 29−35. DOI: https://doi.org/10.1145/345105.345113 − 2000

24. **Kleinbaum D.G., et al.** *Logistic regression*. New York: Springer-Verlag, 2002.

25. **Kalantar B., et al.** Assessment of the effects of training data selection on the landslide susceptibility mapping: A comparison between support vector machine (SVM), logistic regression (LR) and artificial neural networks (ANN). *Geomatics, Natural Hazards and Risk*, 2018, Vol. 9, No. 1, Pp. 49−69.

26. **Ruder S.** An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*. 2016.

СПИСОК ЛИТЕРАТУРЫ

1. **SAP SE.** ABAP—Keyword Documentation // URL: https://help.sap.com/doc/abapdocu_latest_index_htm/latest/en-US/index.htm−2019.

2. **Le Goues C., et al.** Genprog: A generic method for automatic software repair // IEEE Transactions on Software Engineering. 2012. Vol. 38. No. 1. P. 54.

3. **Tan S.H., Roychoudhury A.** relifix: Automated repair of software regressions // Proc. of the 37th Internat. Conf. on Software Engineering. IEEE Press, 2015. Vol. 1. Pp. 471−482.

4. . **Martinez M., Monperrus M.** Astor: A program repair library for java // Proc. of the 25th Internat. Symp. on Software Testing and Analysis. ACM, 2016. Pp. 441−444.

5. **Le X.B.D., Lo D., Le Goues C.** History driven automated program repair. 2016.

6. **Forrest S.** Genetic algorithms: Principles of natural selection applied to computation. Science, 1993, Vol. 261, Pp. 872−878.

7. **Nguyen H.D.T., et al.** Semfix: Program repair via semantic analysis // Proc. of the 35th Internat. Conf. on Software Engineering. IEEE, 2013. Pp. 772−781.

8. **Le X.B.D., et al.** JFIX: semantics-based repair of Java programs via symbolic PathFinder // Proc. of the 26th ACM SIGSOFT Internat. Symp. on Software Testing and Analysis. ACM, 2017. Pp. 376−379.

9. **Wang Y., et al.** CRSearcher: Searching Code Database for Repairing Bugs // Proc. of the 9th Asia-Pacific Symp. on Internetware. ACM, 2017. P. 16.

10. **D'Antoni L., Samanta R., Singh R.** Qlose: Program repair with quantitative objectives // Internat. Conf. on Computer Aided Verification. Springer, Cham, 2016. Pp. 383−401.

11. **Mechtaev S., et al.** Semantic Program Repair Using a Reference Implementation // Proc. of ICSE. 2018.

12. **van Tonder R., Le Goues C.** Static Automated Program Repair for Heap Properties. 2018.

13. **Hill A., Păsăreanu C.S., Stolee K.T.** Automated program repair with canonical constraints // Proc. of the 40th Internat. Conf. on Software Engineering: Companion Proceeedings. ACM, 2018. Pp. 339−341.

14. **Cadar C., et al.** KLEE: Unassisted and Automatic Generation of High-Coverage Tests for Complex Systems Programs // OSDI. 2008. Vol. 8. Pp. 209−224.

15. **De Moura L., Bjørner N.** Z3: An efficient SMT solver // Internat. Conf. on Tools and Algorithms for the Construction and Analysis of Systems. Springer, Berlin, Heidelberg, 2008. Pp. 337−340.

16. **Liu C., et al.** R2Fix: Automatically generating bug fixes from bug reports // Proc. of the 6th Internat. Conf. on Software Testing, Verification and Validation. IEEE, 2013. Pp. 282−291.

17. **Long F., Rinard M.** Automatic patch generation by learning correct code // ACM SIGPLAN Notices. 2016. Vol. 51. No. 1. Pp. 298−312.

18. **Saha R.K., et al.** Elixir: Effective object-oriented program repair // 2017 32nd IEEE/ACM Internat. Conf. on Automated Software Engineering. IEEE, 2017. Pp. 648−659.

19. **Gopinath D., et al.** Data-guided repair of selection statements // Proc. of the 36th Internat. Conf. on Software Engineering. ACM, 2014. Pp. 243−253.

20. **Dietterich T.G.** Machine learning. Encyclopedia of Computer Science. John Wiley and Sons Ltd., GBR, 2003. Pp. 1056−1059.

21. **Witten I.H., et al.** Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, 2016.

22. **Cui B., et al.** Code comparison system based on abstract syntax tree // Proc. of the 3rd IEEE Internat. Conf. on Broadband Network and Multimedia Technology. IEEE, 2010. Pp. 668−673.

23. **Matthew S. Davis.** An object oriented approach to constructing recursive descent parsers // SIGPLAN Not. 2000. 35. 2. Pp. 29−35. DOI: https://doi.org/10.1145/345105.345113 − 2000

24. **Kleinbaum D.G., et al.** Logistic regression. New York: Springer-Verlag, 2002.

25. **Kalantar B., et al.** Assessment of the effects of training data selection on the landslide susceptibility mapping: A comparison between support vector machine (SVM), logistic regression (LR) and artificial neural networks (ANN) // Geomatics, Natural Hazards and Risk. 2018. Vol. 9. No. 1. Pp. 49−69.

26. **Ruder S.** An overview of gradient descent optimization algorithms // arXiv preprint arXiv:1609.04747. 2016.

THE AUTHORS / СВЕДЕНИЯ ОБ АВТОРАХ

Belskii Aleksei
Бельский Алексей
E-mail: belskii.alexey@gmail.com

Itsykson Vladimir M.
Ицыксон Владимир Михайлович
E-mail: vlad@icc.spbstu.ru

SYNTHESIS OF DECENTRALIZED ROBUST
STABILIZING CONTROL FOR THE SYSTEMS
WITH PARAMETRIC PERTURBATIONS

V.N. Kozlov, V.N. Shashikhin

Peter the Great St. Petersburg Polytechnic University,
St. Petersburg, Russian Federation

The paper considers the problem of robust stabilization of large-scale systems with parametric perturbations within set number intervals. Design of systems with robust properties is among the most important problems of control theory. It allows to describe dynamics of the initial object by a mathematical model using a vector differential equation with interval coefficients. The paper states the problem of synthesis of stabilizing control with a pre-assigned degree of robust stability for closed systems. Scalar optimization function and Lyapunov−Razumikhin function were used to identify many stabilizing regulators. Parameters of robust control regulators are determined by the solutions of two Riccati equations: the first one corresponding to the nominal parameters of the object and the second − to variations of the object parameters. Sufficient conditions for robust stability of the closed system are obtained.

Keywords: robust control, large-scale systems, parametric perturbations, decentralized structure, Lyapunov−Razumikhin functions, pre-assigned degree of robust stability.

СИНТЕЗ ДЕЦЕНТРАЛИЗОВАННОГО РОБАСТНОГО
СТАБИЛИЗИРУЮЩЕГО УПРАВЛЕНИЯ СИСТЕМАМИ
С ПАРАМЕТРИЧЕСКИМИ ВОЗМУЩЕНИЯМИ

В.Н. Козлов, В.Н. Шашихин

Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Российская Федерация

Рассмотрена задача робастной стабилизации крупномасштабных систем с параметрическими возмущениями, которые лежат внутри заданных числовых промежутков. Проектирование систем с робастными свойствами является одной из важнейших задач теории автоматического управления. Это позволяет описывать динамику исходного объекта с помощью векторно-матричных дифференциальных уравнений с интервальными коэффициентами. Для класса замкнутых систем сформулирована задача синтеза стабилизирующего управления с заданной степенью робастной устойчивости. С использованием скалярно-оптимизационных функций и функций Ляпунова−Разумихина выделяется множество стабилизирующих регуляторов. Параметры регуляторов определяются решениями уравнения Риккати, связанного с номинальными параметрами объекта, и уравнения Риккати, составленного для вариаций параметров объекта. Получены достаточные условия существования управления, обеспечивающего ро-

бастную стабилизацию исходной системы. Замкнутая система имеет заданную степень робастной устойчивости.

Ключевые слова: робастное управление, крупномасштабные системы, параметрические возмущения, децентрализованная структура, функции Ляпунова−Разумихина, устойчивость с заданной степенью робастности.

Introduction

The defining property of a systems with inexact parameters is the presence of boundaries of a set of initial states and a set of possible trajectories. The systems of this class are of interest not only because of the plentitude of new mathematical problems, but also due to widespread applications of the theory of control of large-scale dynamic systems for the management of both technical objects and economic processes [1–6].

The history of the problem of robustness of the basic dynamic characteristics with respect to various perturbations goes back to the studies of Russian scientists A.A. Andronov and L.S. Pontryagin. To date, a wide range of methods has been developed providing invariance of the characteristics of systems with respect to parametric perturbations based on a variational approach (introducing feedback concerning sensitivity functions), the introduction of a loop with an infinitely large gain, and organization of sliding modes. However, they provide stability only under insignificant variations in parameters.

Modern approaches are focused on ensuring robust stability and quality under significant parametric perturbations. In most cases, these approaches use the robust control synthesis based on solving interval matrix equations of Lyapunov, Sylvester, and Riccati [7, 8, 10–15], or the approaches based on the theory of dynamic games using the strategies of guaranteed result [9, 16].

This paper considers the solution to a robust stabilization problem for systems with parametric perturbations, which are described by interval values of the matrix elements. The author develops an approach to the formation of sufficient conditions for the stability of continuous systems and the synthesis of stabilizing controls with a given measure of robustness, based on using the scalar optimization function of the set and the Lyapunov−Razumikhin interval function [16, 17].

The technique of synthesis of control with robust properties with respect to parametric perturbations uses two-sided target inequalities that determine the requirements for the upper and lower boundaries of the degree of stability of a closed system.

Formulation of the problem

Consider a mathematical model of a large-scale object with parametric perturbations in a form of a system of differential equations with interval coefficients

$$\dot{x} = \tilde{A}x + \tilde{B}u = \left(\tilde{A}_D + \tilde{A}_O \right)x + \tilde{B}u, \, x(0) = x_0, \tag{1}$$

where $x \in \mathbf{R}^n$ is a vector of phase coordinates; $u \in \mathbf{R}^m$ is a vector of control actions; $\tilde{A} \in \mathbf{IR}^{n \times n}$ is an interval matrix of system parameters, $\tilde{A} = \left(\tilde{a}_{ij} \right)_{i,j=1}^n$, $\tilde{a}_{ij} = \left[\underline{a}_{ij}; \overline{a}_{ij} \right] \in \mathbf{IR}$; $\tilde{A}_D = \text{blockdiag} \left\{ \tilde{A}_{ii} \right\}_{i=1}^N$ is a diagonal matrix with the blocks equal to the diagonal blocks of the matrix $\tilde{A}$; $\tilde{A}_O = \text{block} \left\{ \tilde{A}_{ij} \right\}_{i,j=1}^N$ is a non-diagonal

matrix, the blocks of which are non-diagonal blocks of the matrix $\tilde{A}$; $\tilde{B} \in \mathbf{IR}^{m \times n}$ is a matrix of the control action transmission.

The requirements to the dynamic properties of the system are determined by a given degree of stability, which characterizes, on one hand, the location of the eigenvalues of the matrix of the closed system in the open left half-plane, and, on the other hand, the attenuation of some function characterizing the generalized distance from the integral curves of the system to the origin of the phase space coordinates. In this case, it is necessary to provide the required degree of stability in the presence of parametric perturbations.

The degree of stability of the interval system when changing the elements of the matrices $\tilde{A}$ and $\tilde{B}$ is defined by the interval number $\tilde{\alpha} = [\underline{\alpha}; \overline{\alpha}]$.

The index of robust stability is understood as a number e

$$e = 0{,}5 \text{wid } \tilde{\alpha} \, / \, \text{med } \tilde{\alpha},$$

characterizing the relative change in the degree of stability of the interval system. Here $\tilde{\alpha}$ is the width and $\tilde{\alpha}$ is the median of the interval number $\tilde{\alpha}$.

The synthesis of a stabilization system under conditions of parametric uncertainty consists in determining control in the form of feedback with respect to the state vector $u = U(x,t)$, which ensures the fulfillment of target conditions in the form of two-sided inequalities

$$\underline{M}\|x_0\| \exp\left[-\overline{\alpha}(t - t_0)\right] \le \|x(t, t_0, x_0, A, B)\| \le$$
$$\le \overline{M}\|x_0\|\left[-\underline{\alpha}(t - t_0)\right], \tag{2}$$

determining the minimum and maximum rate of the phase coordinate decay in the transition mode for all possible variations of the parameters ($A \in \tilde{A}, B \in \tilde{B}$). Here $\exp\left[-\tilde{\alpha}(t - t_0)\right]$ is the interval function, which is a natural interval extension of the real function $\exp\left[-\alpha(t - t_0)\right]$.

$$\exp\left[-\tilde{\alpha}(t - t_0)\right] = \left[\exp\left[-\overline{\alpha}(t - t_0)\right]; \ \exp\left[-\underline{\alpha}(t - t_0)\right]\right].$$

Introduce the interval Lyapunov function

$$\tilde{V}(x) = x^T \tilde{P} x = \left[x^T \underline{P} x \, ; x^T \overline{P} x\right], \tag{3}$$

which is a natural interval extension of the scalar Lyapunov function, and $\tilde{P} = \left[\underline{P}; \overline{P}\right] \in \mathbf{IR}^{n \times n}$ is a symmetric positive definite interval matrix (any $P \in \tilde{P}$ is a symmetric positive definite matrix). The Lyapunov function (3) allows reformulating the problem of ensuring dynamic stability (2) as the problem of determining control law with feedback that provides a given decay rate of the Lyapunov interval function on the trajectories of perturbed motion $\dot{\tilde{V}}(x) \le -2\tilde{\alpha}\tilde{V}(x)$,

$$\dot{\tilde{V}}(x) \le -2\tilde{\alpha}\tilde{V}(x), \tag{4}$$

where $\tilde{\alpha} = \left[\underline{\alpha}; \overline{\alpha}\right] \in \mathbf{IR}$ is the lower and upper boundaries of the required degree of stability of the closed system; $\dot{\tilde{V}}(x)$ is the derivative of the Lyapunov interval function (3), calculated by virtue of system (1), closed by the regulator $u = U(x,t)$.

Synthesis technique

The essence of the problem of the robust control synthesis lies in the non-uniqueness of the mapping given by the differentiation operator in the system of equations with an indefinitely set right-hand side. To each fixed point (x, θ) of the phase space and the number equal to the value of the positively defined function $V(x, \theta)$ at this point, there is a corresponding set of values of the functional $V(x_t)$ on the set of trajectories x_t coming to this point.

A condition for the effective application of the direct Lyapunov method to the problems of robust stability of the systems with interval coefficients is either the knowledge of the vortices $\{x_t\}$, the segments of the solutions of the system of differential equations that come to this point and correspond to the equivalent initial conditions satisfying the relation $\left\| x_t \left(t_0, x_0, A, B \right) \right\| \le \varepsilon$, or the possibility of constructing estimates based on using scalar optimization functions of the sets of specified vortices.

Using a centered representation of interval matrices $\tilde{A}$ and $\tilde{B}$, the system (1) is reduced to a form [16]

$$\dot{x} = \left(A_0 + \delta \tilde{A} \right) x + \left(B_0 + \delta \tilde{B} \right) u, \quad x(0) = x_0, \tag{5}$$

where $A_0 = \operatorname{med} \tilde{A} \in \mathbf{R}^{n \times n}$, $B_0 = \operatorname{med} \tilde{B} \in \mathbf{R}^{n \times m}$ are the matrices of median (nominal) values of the parameters of the studied system, whereas $\delta \tilde{A} = \left[-\operatorname{rad} \tilde{A}; +\operatorname{rad} \tilde{A} \right] \in \mathbf{IR}^{n \times n}$, $\delta \tilde{B} = \left[-\operatorname{rad} \tilde{B}; + +\operatorname{rad} \tilde{B} \right] \in \mathbf{IR}^{n \times m}$ are interval matrices characterizing the parameter variations.

Denote by $\mu(x(\theta))$ the vortex of integral lines with the vertex at the point (x, θ), which is the set of segments of integral lines of the system of differential equations of the perturbed motion (5), corresponding to the condition $\mu\left(x(\theta) \right) = \left\{ x_t \left| V\left(x(\theta) \right) \le r(L), \theta \ge T \right. \right\}$.

Let us introduce a scalar optimization function $R(x)$ of the set $\mu(x(\theta))$, which establishes a correspondence between the set $\mu(x(\theta))$ and the points of the number axis $\mathbf{R}_+^1$

$$R(x) = \sup \left\{ \dot{V}(x) \left| x(t) \in \mu\left(x(\theta) \right) \right. \right\}. \tag{6}$$

Thus, the scalar optimization function $R(x)$ is determined by the maximum value of the functional $V(x_t)$ on a bounded set of integral curves, along which function $V(x)$ decreases [18].

The condition of exponential stability (4) of the system with parametric perturbations (5) at the scalar optimization function $R(x)$ (6) has the form

$$R(x) \le -\tilde{\alpha} V(x) \tag{7}$$

in the domain

$$\Theta = \left\{ (t, x) \left| t \ge T \ge t_0 + l, \right. \right.$$

$$0 \le V(x) \le r(L) = \sup \left\{ V(x) \left| \|x\| \le L \right. \right\}.$$

Let us construct for the function $\tilde{V}(x)$ a centered form corresponding to the adopted mathematical model of the system with parametric perturbations

$$\tilde{V}(x) = V_0 + \delta \tilde{V} = x^T P_0 x + x^T \delta \tilde{P} x = x^T \left(P_0 + \delta \tilde{P} \right) x, \tag{8}$$

where $P_0 = \mathrm{med}\ \tilde{P} \in \mathbf{R}^{n\times n}$ is the matrix of median values of the quadratic form; $\delta\tilde{P} = \left[-\mathrm{rad}\ \tilde{P}; +\mathrm{rad}\ \tilde{P} \right] = \left[-1; +1 \right] \mathrm{rad}\ \tilde{P} \in \mathbf{IR}^{n\times n}$ is a matrix related to parameter variations.

Now calculate the total derivative of the Lyapunov function (8) by virtue of the system with parametric perturbations (5)

$$\dot{\tilde{V}}(x) = \left(\partial\tilde{V}(x) / \partial x \right)^T (dx / dt) =$$

$$= \dot{x}^T \left(P_0 + \delta\tilde{P} \right) x + x^T \left(P_0 + \delta\tilde{P} \right) \dot{x} \subseteq$$

$$\subseteq x^T \left[\left(A_0 + \delta\tilde{A} \right)^T \left(P_0 + \delta\tilde{P} \right) + \left(P_0 + \delta\tilde{P} \right)\left(A_0 + \delta\tilde{A} \right) \right] x +$$

$$+2x^T \left(P_0 + \delta\tilde{P} \right)\left(B_0 + \delta\tilde{B} \right) u.$$

Then the condition of exponential stability (7) for system (5) is determined by the inequality

$$R(x) = \sup\left\{ x^T \left[\left(A_0 + \delta\tilde{A} \right)^T \left(P_0 + \delta\tilde{P} \right) \right] x + \right.$$

$$+ x^T \left[\left(P_0 + \delta\tilde{P} \right)\left(A_0 + \delta\tilde{A} \right) \right] x +$$

$$\left. +2x^T \left(P_0 + \delta\tilde{P} \right)\left(B_0 + \delta\tilde{B} \right) u \,\middle|\, x \in \mu\left(x(\theta) \right) \right\} \leq$$

$$\leq -\left(\alpha_0 + \delta\tilde{\alpha} \right) x^T \left(P_0 + \delta\tilde{P} \right) x, \tag{9}$$

where $\alpha_0 = \mathrm{med}\ \tilde{\alpha} \in \mathbf{R}^1$, $\delta\tilde{\alpha} = \left[-1; +1 \right] \mathrm{rad}\ \tilde{\alpha}$.

The set of all stabilizing controls that satisfy inequality (9) is described by the relation

$$u = \left\{ -\frac{\left(B_0 + \delta\tilde{B} \right)^T \left(P_0 + \delta\tilde{P} \right) x}{2\left\| x^T \left(P_0 + \delta\tilde{P} \right)\left(B_0 + \delta\tilde{B} \right) \right\|^2} \times \right.$$

$$\left. \times \left[x^T \left[2\left(A_0 + \delta\tilde{A} \right)\left(P_0 + \delta\tilde{P} \right) \right] x + x^T \left[\left(\alpha_0 + \delta\tilde{\alpha} \right)\left(P_0 + \delta\tilde{P} \right) \right] x \right] \right\}. \tag{10}$$

In relation (10), the matrix P_0 satisfies the Riccati matrix equation

$$A_{0\alpha}^T P_0 + P_0 A_{0\alpha} - P_0 F_0 P_0 + Q_0 = 0. \tag{11}$$

Equation (11) corresponds to the nominal system, and the matrices have a form $A_{0\alpha} = A_0 + 0{,}5\alpha_0 I_n$, $F_0 = 10 B_0 B_0^T + 2I_n$, $Q_0 = 2 A_0^T A_0$.

The interval matrix $\delta\tilde{P}$

$$\delta\tilde{P} = \left\{ \delta P \in \mathbf{R}^{n\times n} \,\middle|\, \delta A_\alpha^T \delta P + \delta P \delta A_\alpha - \delta P D \delta P + G = 0, \right.$$

$$\left. \middle|\ \delta A_\alpha \in \delta\tilde{A}_\alpha,\ D \in \tilde{D},\ G \in \tilde{G} \right\}$$

defines the combined set of solutions of the Riccati interval matrix equation

$$\delta \tilde{A}_\alpha^T \delta \tilde{P} + \delta \tilde{P} \delta \tilde{A}_\alpha - \delta \tilde{P} \tilde{D} \delta \tilde{P} + \tilde{G} = 0. \tag{12}$$

Here $\delta \tilde{A}_\alpha = \delta \tilde{A} + 0{,}5 \tilde{\alpha} I_n = [-1; +1] \left(\operatorname{rad} \tilde{A} + 0{,}5 \operatorname{rad} \tilde{\alpha} I_n \right) \in \mathbf{IR}^{n \times n}$, $\tilde{D} = (10 B_0 B_0^T + 4 \delta \tilde{B} \delta \tilde{B}^T - 2 I_n) \in \mathbf{IR}^{n \times n}$, $\tilde{G} = (\delta \tilde{A}^T \delta \tilde{A} - 4 P_0 \delta \tilde{B} \delta \tilde{B}^T P_0) \in \mathbf{IR}^{n \times n}$.

Equation (12) corresponds to variations in the parameters of system (1).

The set of regulators providing robust stability of the interval system (5) with a given indicator e is determined by the following expression

$$u(x) = -B_0^T \left(P_0 + \delta \tilde{P} \right) x = -\left(K_0 + \delta \tilde{K} \right) x, \tag{13a}$$

where $K_0 = B_0^T P_0 \in \mathbf{R}^{m \times n}$, $\delta \tilde{K} = B_0^T \delta \tilde{P} \in \mathbf{IR}^{m \times n}$ or

$$U(x) = \tilde{K} x = \left(\operatorname{med} \tilde{K} \pm 0{,}5 \operatorname{wid} \tilde{K} \right) x, \tag{13b}$$

where $\tilde{K} = -\tilde{B}^T \tilde{P}$ is the matrix of interval feedback coefficients; $\operatorname{med} \tilde{K} = \left(\overline{K} + \underline{K} \right) / 2$ is the median of the interval matrix $\tilde{K}$, setting nominal values of regulator parameters; $\operatorname{wid} \tilde{K} = \left(\overline{K} - \underline{K} \right)$ is the width of the matrix $\tilde{K}$, which determines the maximum margin of the parameters.

Control (13b) is with the matrix $\tilde{P}$, which is a combined set of solutions

$$\tilde{P} = \left\{ P \in \mathbf{R}^{n \times n} \,\middle|\, P(A + \alpha E_n) + (A + \alpha E_n)^T P - \right. \\ \left. -2 P B B^T P = 0, A \in \tilde{A}, B \in \tilde{B}, \alpha \in \tilde{\alpha} \right\}. \tag{14}$$

The matrix $\tilde{P}$ defined by expression (14) satisfies the Riccati interval matrix equation and can be found by the technique presented in [19],

$$\tilde{P} \left(\tilde{A} + \tilde{\alpha} E_n \right) + \left(\tilde{A} + \tilde{\alpha} E_n \right)^T \tilde{P} - 2 \tilde{P} \tilde{B} \tilde{B}^T \tilde{P} = 0. \tag{15}$$

Control (13b) provides robust stabilization of the system $\dot{x} = \left(\tilde{A} - \tilde{B} \tilde{K} \right) x$ with a degree of stability $\tilde{\alpha}$ belonging to the given interval $\left[\underline{\alpha}; \overline{\alpha} \right]$.

The robust regulator (13) has a centralized structure, which requires significant computational costs to determine its parameters due to the need of solving the Riccati interval equation of a sufficiently large dimension. Besides, the regulator is difficult to implement, since it requires information on the state of the entire system to form a control action.

The synthesis of decentralized robust control

To reduce the costs of design and implementation, we will seek a robust regulator in a class of decentralized structures

$$\tilde{K} = \left\{ K = \operatorname{blockdiag} \left\{ K_{ii} \right\}_{i=1}^n \in \mathbf{R}^{m \times n} \,\middle|\, \right. \\ \left. \left| \operatorname{Re} \lambda_i \left(A - BK \right) \right| < 0, i = \overline{1, n}, A \in \tilde{A}, B \in \tilde{B} \right\}. \tag{16}$$

To this end, using the idea of inverse optimization, we carry out a decomposition of the Riccati equation (15), presenting the matrix $\tilde{A}$ as a sum of the block diagonal matrix $\tilde{A}_D$ and the block non-diagonal matrix $\tilde{A}_O$

$$\tilde{P}\left(\tilde{A}_D + \tilde{\alpha}E_n\right) + \left(\tilde{A}_D + \tilde{\alpha}E_n\right)^T \tilde{P} - $$
$$-2\tilde{P}\tilde{B}\tilde{B}^T\tilde{P} + \tilde{P}\tilde{A}_O + \tilde{A}_O^T\tilde{P} = 0. \tag{17}$$

To satisfy structural constraints (16), we introduce into the Riccati equation (17) an additional term $\tilde{Q}_O$, the value of which is determined by the intensity of connections between the subsystems

$$\tilde{Q}_O = -\left(\tilde{A}_O^T\tilde{P} + \tilde{P}\tilde{A}_O\right). \tag{18}$$

Then it becomes possible to decompose equation (17) into N equations, each of which corresponds to the ith subsystem included in the large-scale system (1)

$$\tilde{P}_{ii}\left(\tilde{A}_{ii} + \tilde{\alpha}E_{n_i}\right) + \left(\tilde{A}_{ii} + \tilde{\alpha}E_{n_i}\right)^T \tilde{P}_{ii} - $$
$$-2\tilde{P}_{ii}\tilde{B}_{ii}\tilde{B}_{ii}^T\tilde{P}_{ii} = 0, \, i = \overline{1,N}, \tag{19}$$

whereas matrix $\tilde{P}$ acquires the desired diagonal structure $\tilde{P} = \text{blockdiag}\left\{\tilde{P}_{ii}\right\}_{i=1}^N$. The block elements of the matrix $\tilde{Q}_O$ in relation (18) are determined by the following formula

$$\tilde{Q}_O = \text{block}\left\{\tilde{Q}_{ij}\right\}_{i,j=1}^N = \begin{cases} \tilde{Q}_{ii} = 0, i = j \\ \tilde{Q}_{ij} = \sum_{i=1}^N \left(\tilde{P}_{ii}\tilde{A}_{ij} + \tilde{A}_{ji}^T\tilde{P}_{jj}\right), i \neq j \end{cases} \tag{20}$$

The matrix $\tilde{K}$ of feedback coefficients, due to the block diagonal structure of the matrices $\tilde{B}$ and $\tilde{P}$, is also a block diagonal matrix, and the decentralized robust regulator takes the form

$$u_i\left(x_i\right) = \tilde{K}_{ii}x_i = -\tilde{B}_{ii}^T\tilde{P}_{ii}x_i, i = \overline{1,N}, \tag{21}$$

where the interval matrix

$$\tilde{P}_{ii} = \left\{P_{ii} \in \mathrm{R}^{n_i \times n_i} \, \middle| \, P_{ii}\left(A_{ii} + \alpha E_{n_i}\right) + \left(A_{ii} + \alpha E_{n_i}\right)^T P_{ii} - \right.$$
$$\left. -2P_{ii}B_{ii}B_{ii}^T P_{ii} = 0, \, A_{ii} \in \tilde{A}_{ii}, \, B_{ii} \in \tilde{B}_{ii}, \, \alpha \in \tilde{\alpha}\right\}$$

is a combined set of solutions of the Riccati interval equation (19).

Relation (21) defines the set of stabilizing regulators. Any regulator with the feedback coefficient $K_{ii} \in \tilde{K}_{ii}$ ensures stability of interconnected subsystems

$$\dot{x}_i = \left(\tilde{A}_{ii} + \tilde{B}_{ii}\tilde{K}_{ii}\right)x_i + \sum_{j=1, j\neq i}^N \tilde{A}_{ij}x_j, i = \overline{1,N} \tag{22a}$$

or stability of the large-scale system as a whole

$$\dot{x} = \left(\tilde{A}_D + \tilde{B}\tilde{K} \right) x + \tilde{A}_O x. \tag{22b}$$

Also, as in the case of a centralized regulator, it is expedient to choose the feedback coefficient K_{ii} in the following way

$$K_{ii} = \left(\operatorname{med} \tilde{K}_{ii} \pm 0.5 \operatorname{wid} \tilde{K}_{ii} \right), \ i = \overline{1, N}, \tag{23}$$

thereby determining the nominal parameters of the decentralized regulator $K_{0ii} = \operatorname{med} \tilde{K}_{ii} = \left(\overline{K}_{ii} + \underline{K}_{ii} \right) /$ /2 and the margin for their change $\delta K_{ii} = \operatorname{wid} \tilde{K}_{ii} = \left(\overline{K}_{ii} - \underline{K}_{ii} \right)$.

Properties of the decentralized control

The decentralized control is built in a form of feedback on the phase vector taking into account the mutual influence of subsystems, and ensures the degree of stability of the closed-loop system within a given interval. The use of decentralized control makes it possible to reduce the synthesis costs due to the decomposition of the Riccati matrix equations, which are solved during the synthesis associated with subsystems whose dimensions are less than the dimension of the original large-scale system. The proof of the theorem on the properties of decentralized control is based on a scalar optimization function, which is an analogue of the derivative of the Lyapunov function for systems with parametric perturbations.

The properties of a large-scale system closed by a decentralized regulator are defined in the following theorem.

Theorem. Suppose that for each subsystem included in a large-scale system, the following conditions are fulfilled:

1) rank control criterion

$$\operatorname{rank} \tilde{D}_{ii} = n_i, \ \tilde{D}_{ii} = \left(\tilde{B}_{ii} \left| \tilde{A}_{ii} \tilde{B}_{ii} \right| \ldots \left| \tilde{A}_{ii}^{(n_i-1)} \tilde{B}_{ii} \right. \right), \tag{24}$$

2) existence of the inverse matrix for the test matrix of the form

$$\tilde{W}_i = \left(\left(\tilde{A}_{ii} + \tilde{\alpha} E_{n_i} \right)^T \otimes E_{n_i} + E_{n_i} \otimes \left(\tilde{A}_{ii} + \tilde{\alpha} E_{n_i} \right) \right), \tag{25}$$

where the rank of the test matrix equals n_i^2.

Then the decentralized control (21), including the regulator of the form (23), ensures the stability of the system (22) for all variations of the parameters satisfying conditions (25) and (26), while the degree of stability of the closed system belongs to the interval $\tilde{\alpha} = \left[\underline{\alpha}; \overline{\alpha} \right]$.

Proof of the theorem. The fulfillment of the conditions of the theorem guarantees the existence of an external interval solution $\tilde{P}_{Bii} = \left[\underline{P}_{Bii}; \overline{P}_{Bii} \right]$ of the Riccati equation (19), which includes the combined set of solutions $\tilde{P}_{ii} = \left[\underline{P}_{ii}; \overline{P}_{ii} \right]$.

Then there exists an interval quadratic form

$$\tilde{V}_i \left(x_i \right) = x_i^T \tilde{P}_{Bii} x_i = \left[x_i^T \underline{P}_{Bii} x_i ; x_i^T \overline{P}_{Bii} x_{ii} \right], \tag{26}$$

satisfying two-sided inequality

$$\underline{\lambda}_{\min}\left(\tilde{P}_{Bii}\right)\|x\|^2 \le \tilde{V}_i\left(x_i\right) \le \overline{\lambda}_{\max}\left(\tilde{P}_{Bii}\right)\|x\|^2, \tag{27}$$

where $\underline{\lambda}_{\min}\left(\tilde{P}_{Bii}\right)$ is the lower boundary of the minimum eigenvalue, while $\overline{\lambda}_{\max}\left(\tilde{P}_{Bii}\right)$ is the upper boundary of the maximum eigenvalue of the interval matrix $\tilde{P}_{Bii}$, which are calculated by the formulas

$$\tilde{\lambda}_{\min}\left(\tilde{P}_{Bii}\right) = \left[\lambda_{\min}\left\{\text{med}\,\tilde{P}_{Bii}\right\} - \varepsilon_{\min};\lambda_{\min}\left\{\text{med}\,\tilde{P}_{Bii}\right\} + \varepsilon_{\min}\right], \tag{28}$$

$$\tilde{\lambda}_{\max}\left(\tilde{P}_{Bii}\right) = \left[\lambda_{\max}\left\{\text{med}\,\tilde{P}_{Bii}\right\} - \varepsilon_{\max};\lambda_{\max}\left\{\text{med}\,\tilde{P}_{Bii}\right\} + \varepsilon_{\max}\right]. \tag{29}$$

Here the residual vector ε in (28) and (29) is calculated by the formulas

$$\varepsilon_{\min} = \left\|\,\left|\xi_{\min}\right|\,\right\|_2, \varepsilon_{\max} = \left\|\,\left|\xi_{\max}\right|\,\right\|_2, \left\|\,|\xi|\,\right\|_2 = \left(\sum_1^n |\xi_i|^2\right)^{1/2},$$

$$|\xi| = \text{col}\,|\xi_i|, \left|\tilde{\xi}_i\right| = \max\left\{\left|\underline{\xi}_i\right|, \left|\overline{\xi}_i\right|\right\}.$$

Therefore, the quadratic form (26) is a Lyapunov function of the interval type for system (1) with parametric perturbations. Let us calculate the total derivative of the Lyapunov interval function

$$\tilde{V}(x) = \sum_{i=1}^N \tilde{V}_i\left(x_i\right) = x^T \tilde{P}_B x, \tilde{P}_B = \text{blockdiag}\left\{\tilde{P}_{Bii}\right\}_{i=1}^N$$

due to the system, closed by the decentralized control (21)

$$\dot{\tilde{V}}(x) = 2x^T \tilde{P}_B \left(\tilde{A}_D - \tilde{B}\tilde{B}^T \tilde{P}_B + \tilde{A}_O\right)x \subseteq$$

$$\subseteq x^T \left(\tilde{P}_B \tilde{A}_D + \tilde{A}_D^T \tilde{P}_B - 2\tilde{P}_B \tilde{B}\tilde{B}^T \tilde{P}_B + \tilde{P}_B \tilde{A}_O + \tilde{A}_O^T \tilde{P}_B\right)x.$$

Matrices $\tilde{P}_{Bii}$ are solutions to the matrix Riccati equations (19), which are, in total, equivalent to equation (17) when choosing the matrix $\tilde{Q}_O$ in accordance with (20). Therefore, for the derivative of the Lyapunov function under consideration, the following inequality holds:

$$\dot{\tilde{V}}(x) \le -2\tilde{\alpha}x^T\left(\tilde{P}_B + \tilde{Q}_O\right)x. \tag{30}$$

The relation (30) yields the fulfillment of the inequality

$$\tilde{V}\left(x_0\right)\exp\left[-2\overline{\alpha}\left(t - t_0\right)\right] \le \tilde{V}(x) \le \tilde{V}\left(x_0\right)\exp\left[-2\underline{\alpha}\left(t - t_0\right)\right], \tag{31}$$

which testifies to the exponential stability with respect to function $\tilde{V}(x)$. Using estimate (27), we replace $\tilde{V}(x_0)$ in the right-hand side of the last inequality with the larger quantity $\overline{\lambda}_{\max}\left(\tilde{P}_{Bii}\right)\|x_0\|^2$, and $\tilde{V}(x)$, with the smaller quantity $\underline{\lambda}_{\min}\left(\tilde{P}_{Bii}\right)\|x\|^2$; then for solutions of system (22) with the synthesized decentralized regulator, there holds the estimate from above

$$\left\| x\left(t,\, t_0,\, x_0,\, A, B\right)\right\| \le$$

$$\le \left(\overline{\lambda}_{\max}\left(\tilde{P}_{Bii}\right)/\underline{\lambda}_{\min}\left(\tilde{P}_{Bii}\right)\right)^{1/2}\left\| x_0\right\| \exp\left[-\underline{\alpha}\left(t-t_0\right)\right].$$

(32)

Similarly, replacing the quantity $\tilde{V}\left(x_0\right)$ in the left-hand side of inequality (31) with the smaller quantity $\underline{\lambda}_{\min}\left(\tilde{P}_{Bii}\right)\left\| x_0\right\|^2$, and $\tilde{V}\left(x\right)$, with the larger quantity $\overline{\lambda}_{\max}\left(\tilde{P}_{Bii}\right)\left\| x\right\|^2$, we obtain the following estimate from below for the solution of the system

$$\underline{\lambda}_{\min}\left(\tilde{P}_{Bii}\right)/\left(\overline{\lambda}_{\max}\left(\tilde{P}_{Bii}\right)\right)^{1/2}\left\| x_0\right\| \exp\left[-\overline{\alpha}\left(t-t_0\right)\right] \le \left\| x\left(t, t_0, x_0, A, B\right)\right\|.$$

(33)

Inequalities (32) and (33) indicate the fulfillment of the target condition (2). The solutions of system (1) with any regulator from the family of interval regulators (21), including regulator (23), are exponentially stable. The obtained inequalities for the solutions of system (1), (22) are valid for all variations of the parameters satisfying the conditions of the theorem; therefore, the closed system has the property of robust stability with respect to parametric perturbations.

Conclusion

The considered technique for the synthesis of decentralized control of a large-scale system ensures the stability of perturbed motions in the conditions of the interval setting of subsystem parameters. The parameters of the stabilizing regulator are calculated using the solution of the Riccati interval matrix algebraic equation. The decomposition of the matrix equation, based on the ideas of inverse optimization, allows reducing the design costs using a decentralized structure of the regulator.

The proposed procedure for the synthesis of decentralized control provides stability of a large-scale system in comparison with other methods of decentralized control, taking into account the mutual influence of subsystems. The use of the interval model of the system ensures robust stability with respect to internal parametric disturbances, in contrast to control based on Hardy spaces (H_2, H_∞ – control methods). In comparison with the methods based on V.L. Kharitonov's theorems, the considered method allows not only the synthesis of systems with interval parameters, but also the synthesis of systems with the required properties.

REFERENCES

1. **Levin V.I.** Ustoychivost resheniya optimizatsionnykh zadach v usloviyakh neopredelennosti [Stability of solving optimization problems in the conditions of uncertainty]. *Sistemy upravleniya, svyazi i bezopasnosti* [*Control, Communication and Security Systems*], 2015, No. 4, Pp. 260–276. (rus)

2. **Furtat I.B.** Upravleniye elektroenergeticheskoy setyu s uchetom yeye topologii [Control of the electric power network, taking into account its topology]. *Mekhatronika, avtomatizatsiya, upravleniye* [*Mechatronics, Automation, Control*], 2013, No. 4, Pp. 33–38. (rus)

3. **Filippova V.O.** Zadacha optimalnogo upravleniya s intervalnym parametrom v gilbertovom prostranstve [The optimal control problem with an interval parameter in a Hilbert space]. *Zhurnal vychislitelnoy matematiki i matematicheskoy fiziki* [*Journal of Computational Mathematics and Mathematical Physics*], 2013, Vol. 4 (53), Pp. 389–395. (rus)

4. **Fradkov V.L., Furtat I.B.** Robastnoye upravleniye setyu elektricheskikh generatorov [Robust control of a network of electric generators]. *Avtomatika i telemekhanika* [*Automation and Remote Control*], 2013, No. 11, Pp. 100–113. (rus)

5. **Kozlov V.N., Shashikhin V.N.** Sintez koordiniruyushchego robastnogo upravleniya vzaimosvyazannymi sinkhronnymi generatorami [Synthesis of coordinating robust control of interconnected synchronous generators]. *Elektrichestvo* [*Electricity*], 2009, No. 9, Pp. 20–26. (rus)

6. **Sevodin M.A., Sokolov V.A.** O nechetkoy modeli Valrasa rynka odnogo tovara [About the fuzzy Walras' model of a single product market]. *Moscow Economic Journal*, 2018, No. 5–3, Pp. 32–37. (rus)

7. **Filippova V.O.** Zadacha optimalnogo upravleniya s intervalnym parametrom [The optimal control problem with an interval parameter]. *Dalnevostochnyy matematicheskiy zhurnal* [*Far Eastern Mathematical Journal*], 2017, Vol. 1 (17), Pp. 110–123. (rus)

8. **Dli M.I., Puchkov A.Yu.** Integralno-differentsialnyye uravneniya v strukture nechetkogo filtra Kalmana pri upravlenii slozhnymi tekhnologicheskimi obyektami [Integral-differential equations in the structure of the fuzzy Kalman filter in the management of complex technological objects]. *Programmnyye produkty i sistemy* [*Software Products and Systems*], 2017, No. 4, Pp. 567–573. (rus)

9. **Shashikhin V.N.** Robust control with a two-level structure based on a minimax strategy. *Collection of Scientific Papers of the 22ⁿᵈ International Scientific and Practical Conference*, 2018, Pp. 252–262.

10. **Hashemi B., Dehghan M.** The interval Lyapunov matrix equation: analytical results and an efficient numerical technique for outer estimation of the united solution. *Mathematical and Computer modeling*, 2012, Vol. 55, No. 3–4, Pp. 622–633.

11. **Frommer A., Hashemi B.** Verified error bounds solutions of Sylvester matrix equation. *Linear Algebra and its Application*, 2012, Vol. 436, No. 2, Pp. 405–420.

12. **Dehghani-Madisen M., Hladik M.** Enclosing the solutions set of the parametric generalized Sylvester matrix equation. *International Journal of System Science*, 2019, Vol. 50, No. 11, Pp. 2153–2167.

13. **Dehghani-Madisen M.** On the interval generalized coupled matrix equation. *Electronic Journal of Linear Algebra*, 2018, Vol. 34, No. 1, Pp. 695–717.

14. **Dehghani-Madisen M., Hladik M.** Efficient for enclosing the united solutions set of interval generalized Sylvester matrix equation. *Applied Numerical Mathematics*, 2018, Vol. 126, No. 4, Pp. 18–33.

15. **Dehghani-Madisen M., Dehghan M.** Parametric AE- solutions set parametric matrix equation. *Numerical Algorithms*, 2016, Vol. 73, No. 1, Pp. 245–279.

16. **Shashihin V.N.** Guaranteed-result strategy to synthesize a decentralized control for system under parametric perturbations. *Automatic Control and Computer Sciences*, 2017, Vol. 51, No. 2, Pp. 75–84.

17. **Andreev A.S., Sedova N.O.** The method of Lyapunov-Razumikhin functions in stability analysis of systems with delay. *Automation and Remote Control*, 2019, Vol. 80, No. 7, Pp. 1185–1229.

18. **Shashihin V.N.** Synthesis of control for nonlinear systems. *Automatic Control and Computer Sciences*, 2019, Vol. 53, No. 2, Pp. 97–106.

19. **Haqiri T., Morseni Moghadam M., Rivaz A.** The united stable set of interval continuous-time algebraic Riccati equation and verified numerical computation of its outer estimation. *Turkish Journal of Mathematics*, 2018, Vol. 42, No. 3, C. 1130–1155.

СПИСОК ЛИТЕРАТУРЫ

1. **Левин В.И.** Устойчивость решения оптимизационных задач в условиях неопределенности // Системы управления, связи и безопасности. 2015. № 4. С. 260–276.

2. **Фуртат И.Б.** Управление электроэнергетической сетью с учетом ее топологии // Мехатроника, автоматизация, управление. 2013 № 4. С. 33–38.

3. **Филиппова В.О.** Задача оптимального управления с интервальным параметром в гильбертовом пространстве // Журнал вычислительной математики и математической физики. 2013. Т. 53. № 4 С. 389–395.

4. **Фрадков В.Л., Фуртат И.Б.** Робастное управление сетью электрических генераторов // Автоматика и телемеханика. 2013. № 11. С. 100–113.

5. **Козлов В.Н., Шашихин В.Н.** Синтез координирующего робастного управления взаимосвязанными синхронными генераторами // Электричество. 2009. № 9. С. 20–26.

6. **Севодин М.А., Соколов В.А.** О нечеткой модели Вальраса рынка одного товара // Московский экономический журнал. 2018. № 5–3. С. 32–37.

7. **Филиппова В.О.** Задача оптимального управления с интервальным параметром // Дальневосточный математический журнал. 2017. Т. 17. № 1. С. 110–123.

8. **Дли М.И., Пучков А.Ю.** Интегрально-дифференциальные уравнения в структуре нечеткого фильтра Калмана при управлении сложными технологическими объектами // Программные продукты и системы. 2017. № 4. С. 567–573.

9. **Шашихин В.Н.** Робастное управление с двухуровневой структурой на основе минимаксной стратегии // Сб. науч. трудов XXII междунар. науч.-практ. конф. 2018. С. 252–262.

10. **Hashemi B., Dehghan M.** The interval Lyapunov matrix equation: analytical results and an efficient numerical technique for outer estimation of the united solution // Mathematical and Computer modeling. 2012. Vol. 55. No. 3–4. Pp. 622–633.

11. **Frommer A., Hashemi B.** Verified error bounds solutions of Sylvester matrix equation // Linear Algebra and its Application. 2012. Vol. 436. No. 2. Pp. 405–420.

12. **Dehghani-Madisen M., Hladik M.** Enclosing the solutions set of the parametric generalized Sylvester matrix equation // Internat. J. of System Science. 2019. Vol. 50. No. 11. Pp. 2153–2167.

13. **Dehghani-Madisen M.** On the interval generalized coupled matrix equation // Electronic Journal of Linear Algebra. 2018. Vol. 34. No. 1. Pp. 695–717.

14. **Dehghani-Madisen M., Hladik M.** Efficient for enclosing the united solutions set of interval generalized Sylvester matrix equation // Applied Numerical Mathematics. 2018. Vol. 126. No. 4. Pp. 18–33.

15. **Dehghani-Madisen M., Dehghan M.** Parametric AE- solutions set parametric matrix equation // Numerical Algorithms. 2016. Vol. 73. No. 1. Pp. 245–279.

16. **Shashihin V.N.** Guaranteed-result strategy to synthesize a decentralized control for system under parametric perturbations // Automatic Control and Computer Sciences. 2017. Vol. 51. No. 2. Pp. 75–84.

17. **Andreev A.S., Sedova N.O.** The method of Lyapunov-Razumikhin functions in stability analysis of systems with delay // Automation and Remote Control. 2019. Vol. 80. No. 7. Pp. 1185–1229.

18. **Shashihin V.N.** Synthesis of control for nonlinear systems // Automatic Control and Computer Sciences. 2019. Vol. 53. No. 2. Pp. 97–106.

19. **Haqiri T., Morseni Moghadam M., Rivaz A.** The united stable set of interval continuous-time algebraic Riccati equation and verified numerical computation of its outer estimation // Turkish Journal of Mathematics. 2018. Vol. 42. No. 3. Pp. 1130–1155.

THE AUTHOR / СВЕДЕНИЯ ОБ АВТОРЕ

Kozlov Vladimir N.
Козлов Владимир Николаевич
E-mail: umo@spbstu.ru

Shashikhin Vladimir N.
Шашихин Владимир Николаевич
E-mail: shashihin@bk.ru

ИНФОРМАТИКА, ТЕЛЕКОММУНИКАЦИИ И УПРАВЛЕНИЕ

COMPUTING, TELECOMMUNICATIONS AND CONTROL

Том 13, № 2, 2020