Software for energy efficient control system over process parameters of MIMO objects on a set of functioning states

A N Gribkov¹, D Yu Muromtsev¹, V N Shamkin¹, O A Belousov¹, M P Belyaev²

¹Tambov State Technical University, 106, Sovetskaya str., Tambov, 392000, Russia
²The Zhukovsky-Gagarin Air Force Academy, 133, Marshala Nedelina ul, Voronezh, 394052, Russia

E-mail: mdjur@mail.ru

Abstract. Theoretical and practical aspects of building software architecture for energy efficient control of MIMO systems on a set of functioning states are considered. The control systems are classified depending on the nature of changes in the functioning state variable and the possibility of its identification on the control interval. The specifics of software implementation for the systems of various classes are shown. The software is structured with the allocation of functional subsystems and modules. The generalized software architecture for energy efficient control systems on a set of functioning states is proposed.

1. Introduction
Currently, energy efficient control systems (EECS) are used in virtually all industries. The use of the EECSs not only significantly reduces energy costs and increases the productivity of process units, but also ensures the required quality of products. In this regard, issues related to the development of such systems are very relevant and attract considerable attention of researchers.

From the perspective of automation, most modern industrial installations are complex MIMO systems that have a lot of interrelated input and output variables [1, 2]. Such systems include multipurpose vehicles in which phase composition changes, structural transitions, etc. occur. One of the most widespread and energy-consuming MIMO systems are heat engineering equipment, for example, drying plants, industrial furnaces, boiler plants, etc.

At present, a great deal of papers and monographs on the problems of designing automation and control systems have been published. The works of many scientists working in this subject area are known (e.g. L.S. Pontryagin, V.G. Boltyansky, RV Gamkrelidze, E.F. Mishchenko, R. Bellman, A.M. Letov, A.A. Krassovsky, A.M. Lyapunov, M. Atans, P. Falb., A. Hurwitz, H. Nyquist, A.V. Mikhailov, V.A. Trapeznikov, N.N. Krasovsky, V.A. Besekersky, A.G. Aleksandrov, V.V. Solodovnikov, A.A. Voronov, K.A. Pupkov, N.D. Eguopov, V.I. Zubov, V.S. Pugachev, B.T. Polyak, I.P. Norenkov, G.K. Goodwin, S.F. Grebe, M.E. Salgado, J. Lanning, R. Betting, D. Newton, L. Gould, D. Kaiser, A.A. Feldbaum, Yu.L. Muromtsev, A.M. Tsykunov, E.A. Parsheva, E.L. Eremin, A.Z. Asanov, etc.)

Many Russian and foreign scientists are working on the problem of building intelligent control systems. This is due to a number of reasons, the most important of which are: the continuous complication of control objects; the emergence of high-performance computer facilities; a significant increase in the requirements for the accuracy, reliability, and efficiency of control algorithms under uncertainty; the need for synthesis of optimal control actions in real time; the complexity of obtaining an exact mathematical model of the object, etc.
2. Statement of the problem
The process of developing the ECCS for MIMO systems is quite laborious and knowledge-intensive both at the stage of creating the algorithmic support of the system and at the stage of software implementation of the created algorithms [3]. This is because the development of the algorithmic support of the system requires a rather complex mathematical apparatus based on the methods of optimal control theory, decision-making under uncertainty, artificial intelligence, etc. In turn, the algorithmic complexity of the system leads to a complex software architecture of the EECS, which must ensure the possibility for a quickly control over the process system in real time.

When designing the software for the EECS, it is necessary to take into account that in the course of its further real operation, changes in the parameters of the technological process, environment, intensity of disturbing effects, etc., might occur. These changes can have a significant impact on the accomplishment of the goal of control, so the system must take them into account and react quickly to them.

In this connection, the task of designing the software architecture of energy efficient control for MIMO systems, taking into account the possible changes in the functioning states in the process of real operation, is quite relevant.

3. Theoretical grounds
The effectiveness of any system is determined by its properties (reliability, accuracy, speed, etc.), working conditions, the influence of external disturbances and other factors that can be either deterministic or random. For the complex accounting of the main factors affecting the operation of the EECS, the concept of a set of functioning states (SFS) is introduced [4].

When considering the EECS on the SFS, a variable \( h \) is used to indicate the current state of functioning. In this case \( h \in \mathbf{H} \), where \( \mathbf{H} \) is a SFS, i.e. a set of possible values of the variable \( h \). The change in the value of \( h \) can occur at certain or random times. The factors influencing the change in the value of the variable \( h \) may include the changes in the ambient temperature, atmospheric pressure, humidity, process parameters, and other factors that may affect the system efficiency and the accomplishment of the control goal. In general, the SFS is determined by the design features of the EECS for a particular process facility. Depending on the nature of changes in the variable \( h \) and the possibility of its identification on the control interval, four main classes of systems in the SFS are distinguished [4].

The Class 1 EECSs on the SFS are characterized by the fact that the value of the state variable \( h \) is known before the start of the control and it remains constant on the control interval. An example of the first class system can be a reliable control system for simple batch process plants, in which changes in the functioning state occur outside the control interval, while the interval itself is usually short.

The Class 2 EECSs on the SFS are characterized by the fact that the value of the state variable \( h \) before the start of control is unknown and does not change over the control interval. In this case, the value \( h \) is also unknown on the control interval, but probabilities of possible values of the variable \( h \) may be known. The examples of systems of the second class are stochastic systems and systems with differential inclusions.

The Class 3 EECSs on the SFS are characterized by the fact that the value of the state variable \( h \) before the start of control is known and it can change over control interval, with the value of \( h \) at each time instant being identifiable with high accuracy. Dynamic modes of the third class systems are conveniently described with the help of “multistage” models in the form of differential controls systems with a discontinuous right-hand side.

The Class 4 EECSs on the SFS are characterized by the fact that the value of the state variable \( h \) before the start of control can be either known or unknown and it can change over control interval, but it cannot be accurately identified. When designing systems of the fourth class, their algorithms can include models that make it possible to simulate possible trajectories of changing the state variable \( h \) in the course of actual operation.
An important condition for the effective functioning of Classes 1-4 systems is the fulfillment of inclusion conditions. It should also be noted that the above classification of the EECS on the SFS does not show all possible options for the systems that are used in practice. In addition to the four classes identified, various intermediate options are also possible, depending on conditions of actual operation. For example, one and the same system can belong to different classes depending on the mode of operation.

4. Experimental
One of the most important stages in the design of the EECS software architecture is its structuring, i.e. allocation of subsystems and modules. Obviously, the composition of the subsystems and modules, as well as the complexity of their software implementation, will largely be determined by the class to which the designed system belongs to on the SFS. From the above classification of systems on the SFS, it can be seen that some classes are special cases of others, while the EECSs belonging to third and fourth classes are the most complicated in terms of software implementation.

For MIMO systems, the generalized software architecture is presented in Figure 1. The generalized architecture of the system includes a knowledge control subsystem, an interface subsystem and six program modules (in Figure 1, to the left of the software module name, the classes of systems that can include this module are listed).

The knowledge control subsystem includes:
- the knowledge base, in which procedural knowledge that implements algorithmic support of the EECS is stored (the knowledge base can be implemented using an object-oriented approach based on the frame and product knowledge models [5]);
- the output machine, which is a software module that manages knowledge;
- the relational database containing arrays of source data and the outcomes of the EECS operation;
- the database management system (DBMS), providing the necessary interface to work with the database.

The generalized software architecture for the EECS

| Program modules                                      | Knowledge control subsystem                      |
|------------------------------------------------------|--------------------------------------------------|
| 1-4: Module for model dynamics identification        | Knowledge database                               |
| 1-4: Module for analysis of optimal control problems | Output machine                                    |
| 1-4: Module of control actions synthesis             | DBMS                                             |
| 2-4: Simulation module                               | Database                                          |
| 3: Module for identification of currents functioning state |                                         |
| 3,4: Module for experiment planning                  | Module for source data input                     |
|                                                    | Module for cognitive graphics                     |
|                                                    | Integrated development environment                |

**Figure 1.** The software architecture for the EECS of Classes 1-4 on the SFS

The interface subsystem includes:
- the module for source data input;
- the module for cognitive graphics that provides a visual interface for displaying the results of the EECS functioning (cognitive models for solving energy efficient control problems [6], mimic diagrams of the control object, etc.).
– the integrated development environment that allows making changes to the knowledge base and configuring the software modules of the EECS (the need for making changes arises when the process equipment is changed or its operating modes are changed, the product range is changed, etc.).

The software modules included in the generalized architecture of the EECS have the following functional purpose: [7]

– the module for model dynamics identification solves the problem of parametric identification of model dynamics of the MIMO system [8];

– the module for analysis of optimal control problems is used to quickly obtain the results of a complete analysis of the optimal control problem, i.e. to study the domain of existence of the solution of the problem, determine the types of optimal control functions and relationships for calculating their parameters [9, 10];

– the module for control actions synthesis provides the choice of the optimal control algorithm for the system, while the results obtained during the operation of other modules of the system are used.

– the simulation module is designed to compare the effectiveness of control algorithms with the possible changes in the functioning state variable of the control object (the results can later be used to select the most effective control algorithm in real operation);

– the module for the functioning state identification is used in Class 3 systems and is designed to determine the current functioning state of the control object, using the data received through the measurement channel (different types of sensors and measuring devices, for example, temperature, humidity sensors, etc. can be used to obtain data);

– the module for experiment planning provides an experimental plan and statistical data for a given number of experiments (usually applied in conjunction with the simulation module in Class 3 and Class 4 systems);

The software modules of the EECS exchange data with each other using a database in which the results of operation of all the system modules are stored.

5. Results and discussion

The proposed modular architecture of the control system allows taking into account the design features of the EECS on the SFS. It is obvious that the EECSs of different classes will differ from each other by the presence or absence of a number of software modules. For example, for the functioning of EECSs of Class 1, it is sufficient to have the three upper program modules shown in Fig. 1, while the systems of the class 2 must be complemented with a simulation module, etc.

At the same time, the proposed structuring of the system into subsystems and modules is sufficiently universal and invariant for various MIMO systems. At the same time, in spite of the general functional purpose of the program modules considered for systems of different classes, their immediate software implementation can vary significantly depending on the complexity of the particular control object for which the EECS is designed.

In the practical development of the system software modules, it is possible to use both specialized instrumental software, for example, SCADA systems, and the application development tools based on CASE and RAD technologies. The first approach is very convenient for the development of the EECS on the basis of the existing industrial management system and makes it possible, without creating a system from scratch, to add new modules or expand the functionality of the modules already available in the system.

6. Conclusion

The features of designing the software architecture of the EECS for MIMO systems have been considered. The problem of structuring the software of the EECS with the allocation of functional subsystems and modules has been solved. A generalized software architecture for the EECS on the SFS, which allows developing control systems of various classes, has been proposed.

This approach has been applied in the development of software architecture of energy efficient control systems for the dynamic modes of roller-drum and drum-type dryer installations. The developed
control systems have made it possible to reduce energy costs in dynamic operating modes by 5-10% and provide the required values of the product quality indicators [11, 12].

Acknowledgements

The work was supported by the Russian Foundation for Basic Research, Grants No. 17-08-00457-a, No. 18-08-00555-a.

References

[1] Goodwin G C, Graebe S F, and Salgado ME 2000 *Control system design* (New Jersey: Prentice Hall)

[2] Khalil H K 1996 Adaptive output feedback control of nonlinear systems represented by input-output models *IEEE Trans. Automat. Control* 41(2) 177-188

[3] Muromtsev D Y, Gribkov A N, Shamkin V N, Divin A G, and Savenkov A P 2017 Functional modeling of business processes for development of control and monitoring systems *Proceedings of the 2017 International Conference “Quality Management, Transport and Information Security, Information Technologies” (IT&QM&IS)* pp 440-442

[4] Muromtsev Yu L, Ljapin L N and Popova O V 1992 Modelirovanie i optimizacij slozhnyh sistem pri izmenenijah sostojanija funkcionirova-nija (Voronezh : Izd-vo VGU)

[5] Muromtsev D Y, Gribkov A N and Kurkin I A 2015 Baza znaniij intellektual’noj informacionno-upravljaushjhej sistemy mnogomer-nym ob’ekтом *Informacionno-upravljaushhie sistemy* 5 60-64

[6] Muromtsev D Y and Pogonin V A 2015 Specificity of analysis and synthesis of energy efficient control *Jour. of Eng. and App. Sciences* 10(3) 37-39

[7] Gribkov A N and Kurkin I A 2013 Struktura programmного обесpechenija informacionno-upravljaushhii sistem mnogomernymi ob’ekтами на mnozhestve sostojanij funkcionirovaniia Novye informacionnye tehnologii v nauchnyh issledovanijah: mat-ly XVIII Vserossijskoj nauch.-tehn. konf. studentov, molodyh uche-nyh i specialistov. (Rjazan’: Rjazanskij gosudarstvennyj radiotehnicheskij universitet) pp 244-245

[8] Muromtsev D Y, Gribkov A N and Kurkin I A 2014 Certificate of computer programs of state registration 2014611295 of 30.01.2014 “Programmnyj modul’ parametricheskoj identifikacii matematischeskikh modelej dinamiki mnogosvjaznyh ob’ekтов na mnozhestve sostojanij funkcionirovaniia”

[9] Pchelintsev A N and Pogonin V A 2015 A method of determining the control parameters in the energy-saving control problem *Jour. of Eng. and Appl.Sciences* 10(2) 33-36

[10] Gribkov A N and Kurkin I A 2011 Analitiko-graficheskij metod issledovanija oblasti sushhhestvovaniia reshenija zadachi optimal'nogo upravlenija mnogomernym ob’ekтом *Informatika i sistemy upravlenija* 3(29) 141-152

[11] Gribkov A N 2009 Informacionno-upravljaushhaja sistema dinamicheskimi rezhimami v mnogosekcionnyh sushii nых ustanovkah *Informatika i sistemy upravlenija* 2(20) 123-129

[12] Gribkov A N, Kurkin I A and Gavrilov A O 2016 Programmnoe obespechenie informacionno-upravljaushhjej sistemy dinamicheskimi rezhimami processov sushi poslespirtovoj bardy *Aktual'nye problemy jenergosberezhenija i jenergojeffektivnosti v tehnichestkih sistemah: tezisy dokladov 3-oj Mezhdunar. konf. s jelementami nauchnoj shk* (Tambov: Izd-vo Pershina R.V.) pp 61-62