

УДК 004.415.23  
DOI: 10.15827/0236-235X.124.786-790

Дата подачи статьи: 06.07.18  
2018. Т. 31. № 4. С. 786–790

## **Методика построения архитектуры ПО систем энергосберегающего управления многомерными технологическими объектами на множестве состояний функционирования**

А.Н. Грибков<sup>1</sup>, д.т.н., доцент, GribkovAlexey@yandex.ru

Д.Ю. Муромцев<sup>1</sup>, д.т.н., профессор, mdjur@mail.ru

В.Н. Шамкин<sup>1</sup>, д.т.н., доцент, shamkin-v@mail.ru

И.В. Тюрин<sup>1</sup>, к.т.н., доцент, tyrinilja@yandex.ru

<sup>1</sup> Тамбовский государственный технический университет, г. Тамбов, 392000, Россия

В статье рассмотрены теоретические и практические аспекты построения архитектуры ПО систем энергосберегающего управления многомерными технологическими объектами на множестве состояний функционирования. Приведена классификация систем управления, учитывающая изменения состояния функционирования в процессе реальной эксплуатации, а также возможности идентификации текущего состояния функционирования на временном интервале управления. Проведено структурирование ПО систем энергосберегающего управления с выделением функциональных подсистем и программных модулей.

Рассмотрена общая архитектура системы энергосберегающего управления, включающая подсистему управления знаниями и данными, подсистему интерфейса и шесть базовых программных модулей. В состав подсистемы управления знаниями и данными включены база знаний, машина вывода, БД и система управления БД.

Подсистема интерфейса состоит из модуля ввода исходных данных, модуля когнитивной графики и интегрированной среды разработки. В состав базовых программных модулей системы энергосберегающего управления входят модуль идентификации модели динамики объекта управления, модуль анализа задач оптимального управления, модуль синтеза оптимальных управляющих воздействий, модуль имитационного моделирования, модуль идентификации текущего состояния функционирования и модуль планирования эксперимента. Для каждого базового модуля определены классы систем на множестве состояний функционирования, в состав архитектуры ПО которых он может быть включен.

Предложена методика построения архитектуры ПО систем энергосберегающего управления различных классов на множестве состояний функционирования.

Рассмотрены особенности программной реализации систем энергосберегающего управления на базе различных подходов (с использованием прикладных и инструментальных программных средств).

**Ключевые слова:** программное обеспечение, программная архитектура, система энергосберегающего управления, многомерный технологический объект, множество состояний функционирования.

Во многих отраслях промышленности системы энергосберегающего управления (СЭУ) находят широкое применение. Практическое применение СЭУ на промышленных предприятиях позволяет не только существенно снизить затраты энергоресурсов и повысить производительность технологических установок, но и обеспечить требуемые параметры и качественные характеристики выпускаемой продукции. Поэтому так актуальны вопросы, связанные с разработкой таких систем.

К настоящему времени вопросам проектирования и построения различных систем автоматизации и управления технологическими объектами и процессами посвящено немало научных статей и монографий. Однако в данной предметной области имеется множество вопросов, которые требуют проведения дальнейших исследований, что обусловлено не только актуальностью, но и высокими темпами развития современных информационных и компьютерных технологий.

С точки зрения автоматизации современные промышленные технологические установки, как правило, представляют собой сложные многомерные многосвязные объекты, имеющие множество

входных и выходных переменных [1, 2]. Примерами таких объектов являются теплотехнологические аппараты – сушильные установки, промышленные печи, котельные установки и др. Разработка СЭУ многомерными технологическими объектами во многих случаях представляет собой сложное научно-техническое исследование, связанное с детальной проработкой вопросов, связанных с проектированием алгоритмического обеспечения системы и программной реализацией полученных алгоритмов. Следует также учитывать, что при разработке алгоритмического обеспечения системы во многих случаях применяется достаточно сложный математический аппарат, основанный на методах теории оптимального управления, принятия решений в условиях неопределенности, искусственного интеллекта и т.д. Алгоритмическая сложность системы приводит к сложной программной архитектуре СЭУ.

При проектировании ПО СЭУ необходимо учитывать, что в процессе ее реальной эксплуатации могут происходить изменения параметров технологического объекта, окружающей среды, интенсивности возмущающих воздействий и т.д. Эти из-

менения могут оказывать существенное влияние на достижение цели управления, поэтому система должна их учитывать и оперативно реагировать. В связи с этим весьма актуальной является задача построения архитектуры ПО СЭУ многомерными технологическими объектами с учетом возможных изменений состояния функционирования в процессе реальной эксплуатации.

### Классификация систем управления

Эффективность функционирования системы управления во многом определяется ее свойствами (надежность, точность, быстродействие и т.д.), условиями работы, влиянием внешних возмущений и другими факторами, которые могут быть как детерминированными, так и случайными. Для комплексного учета основных факторов, оказывающих влияние на работу СЭУ, вводится понятие *множества состояний функционирования* (МСФ) [3].

При рассмотрении СЭУ на МСФ используется переменная  $h$ , обозначающая текущее состояние функционирования. При этом  $h \in H$ , где  $H$  – МСФ, то есть множество возможных значений переменной  $h$ . Изменения значения  $h$  могут происходить в известные или случайные моменты времени. Значение переменной  $h$  зависит от изменения температуры окружающей среды, атмосферного давления, влажности, параметров технологического процесса и других факторов, которые могут повлиять на эффективность функционирования СЭУ и достижение цели управления. В целом МСФ определяется особенностями построения СЭУ для конкретного технологического объекта. В зависимости от характера изменений переменной  $h$  и возможности ее идентификации на временном интервале управления выделяют четыре основных класса систем на МСФ [3].

СЭУ на МСФ первого класса характеризуются тем, что значение переменной состояния функционирования  $h$  известно до начала управления и на временном интервале управления остается постоянным. Примером системы первого класса может быть надежная система управления простыми технологическими установками периодического действия, в которой изменения состояния функционирования происходят вне временного интервала управления, при этом сам интервал, как правило, небольшой.

СЭУ на МСФ второго класса характеризуются тем, что значение переменной состояния функционирования  $h$  до начала управления неизвестно и на временном интервале управления не меняется. При этом значение  $h$  не может быть точно идентифицировано на временном интервале управления, но могут быть известны вероятности возможных значений переменной  $h$ . Примерами систем второго класса являются стохастические системы и системы с дифференциальными включениями.

СЭУ на МСФ третьего класса характеризуются тем, что значение переменной состояния функционирования  $h$  до начала управления известно и на временном интервале управления может меняться, при этом значение  $h$  в каждый момент времени может быть идентифицировано с высокой точностью. Динамические режимы систем третьего класса удобно описывать с помощью «многостадийных» моделей в виде систем дифференциальных уравнений с разрывной правой частью.

СЭУ на МСФ четвертого класса характеризуются тем, что значение переменной состояния функционирования  $h$  до начала управления может быть известно или неизвестно и на временном интервале управления может меняться, но не может быть точно идентифицировано. При проектировании систем четвертого класса в состав их алгоритмического обеспечения могут включаться модели, позволяющие осуществлять имитационное моделирование возможных траекторий изменения переменной состояния функционирования  $h$  в процессе реальной эксплуатации.

Важным для эффективного функционирования систем третьего и четвертого классов является выполнение условий включаемости [4]. Следует также отметить, что рассмотренная классификация СЭУ на МСФ не включает все возможные варианты систем, которые встречаются на практике. Помимо выделенных основных четырех классов систем, возможны различные частные случаи, когда систему в условиях реальной эксплуатации можно в одних режимах работы отнести к одному классу, а в других – к другому. Следует также отметить, что подход, основанный на рассмотрении систем на МСФ, находит широкое применение на практике, в частности, этому посвящены работы [5–7].

### Архитектура ПО для систем управления различных классов

В качестве технического обеспечения современных систем управления широко применяются персональные компьютеры или промышленные контроллеры, обладающие высокой вычислительной мощностью и производительностью, поэтому ПО СЭУ используется не только для реализации разработанных алгоритмов анализа и синтеза оптимального управления, но и для хранения результатов функционирования системы в БД, обеспечения наглядного и удобного интерфейса пользователя и т.д.

В силу этого одним из важнейших этапов построения архитектуры ПО СЭУ является ее структурирование, то есть выделение подсистем и модулей. Очевидно, что состав подсистем и модулей, а также сложность их программной реализации во многом будут определяться тем, к какому классу на МСФ относится проектируемая СЭУ. Из приведен-

ной выше классификации систем на МСФ видно, что одни классы являются частными случаями других, при этом наиболее сложными с точки зрения программной реализации являются СЭУ третьего и четвертого классов.

Для СЭУ многомерными технологическими объектами предлагается общая архитектура ПО, показанная на рисунке. Система включает подсистему управления знаниями и данными, подсистему интерфейса и шесть программных модулей.

В состав подсистемы управления знаниями и данными входят:

- база знаний, в которой хранятся процедурные знания, реализующие алгоритмическое обеспечение СЭУ (программно база знаний может быть реализована с использованием объектно-ориентированного подхода на базе фреймворка и продукционной модели знаний);
- машина вывода, представляющая собой программный модуль, управляющий знаниями;
- БД, содержащая массивы исходных данных и результаты функционирования СЭУ;
- СУБД, обеспечивающая необходимый интерфейс для работы с БД.

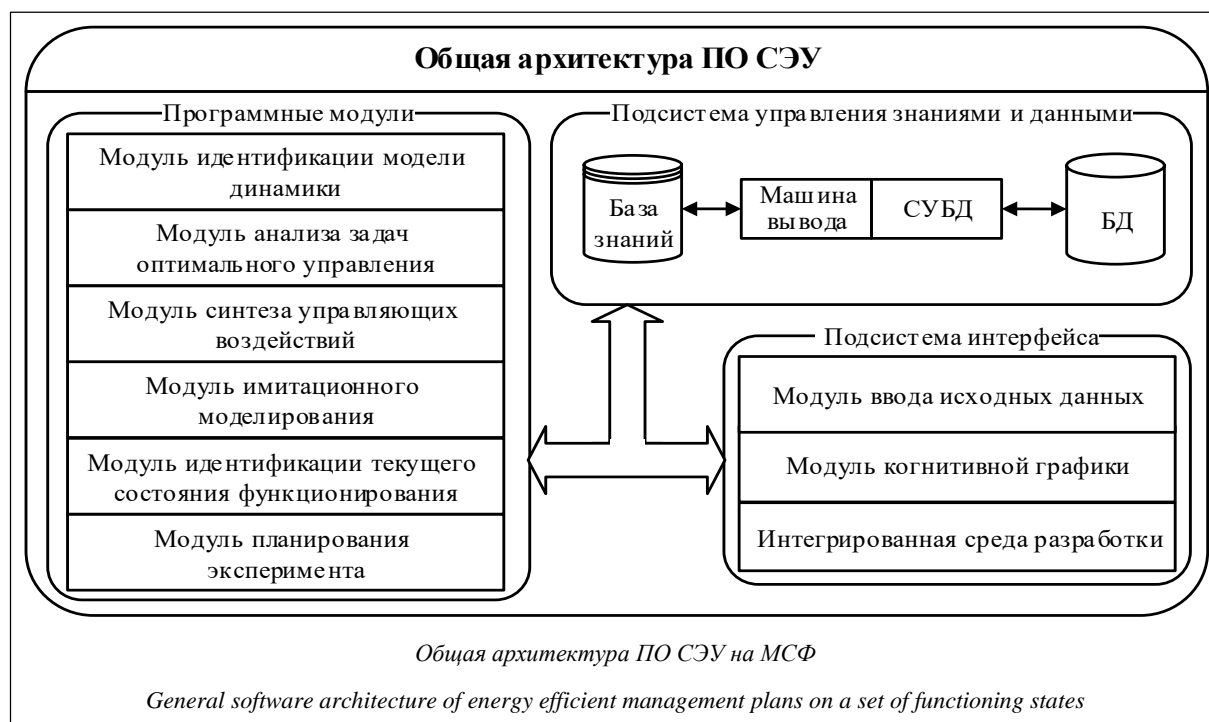
Подсистема интерфейса включает:

- модуль ввода исходных данных;
- модуль когнитивной графики, обеспечивающий наглядный интерфейс для отображения результатов функционирования СЭУ (когнитивные модели решения задач энергосберегающего управления [8], мнемосхемы объекта управления и т.д.);
- интегрированную среду разработки, позволяющую вносить изменения в базу знаний и осуществлять настройку программных модулей СЭУ (не-

обходимость внесения изменений возникает при замене технологического оборудования или изменении режимов его работы, смене номенклатуры выпускаемой продукции и т.д.).

Программные модули, входящие в состав обобщенной архитектуры СЭУ [9], имеют следующее функциональное назначение:

- модуль идентификации модели динамики обеспечивает решение задачи параметрической идентификации модели динамики многомерного объекта (может применяться в системах всех четырех классов);
- модуль анализа задач оптимального управления применяется для оперативного получения результатов полного анализа задачи оптимального управления, то есть исследования области существования решения задачи, определения видов функций оптимального управления и соотношений для расчета их параметров (может применяться в системах всех четырех классов);
- модуль синтеза управляющих воздействий обеспечивает выбор оптимального алгоритма управления объектом, при этом используются результаты, полученные в процессе работы других модулей системы (может применяться в системах всех четырех классов);
- модуль имитационного моделирования предназначен для сравнения эффективности алгоритмов управления с учетом возможных изменений переменной состояния функционирования объекта управления (полученные результаты в дальнейшем могут использоваться при выборе наиболее эффективного алгоритма управления объектом в системах третьего и четвертого классов);



– модуль идентификации состояния функционирования применяется в системах третьего класса и предназначен для определения текущего состояния функционирования объекта управления по данным, получаемым по каналу измерения (для получения данных могут применяться различные типы датчиков и измерительных устройств, например, датчики температуры, влажности и т.д.);

– модуль планирования эксперимента обеспечивает составление плана эксперимента и получение статистических данных по заданному количеству опытов (применяется, как правило, совместно с модулем имитационного моделирования в системах третьего и четвертого классов).

Программные модули СЭУ обмениваются друг с другом данными при помощи БД, в которой хранятся результаты работы всех модулей системы.

### Методика построения архитектуры ПО СЭУ на множестве состояний функционирования

Рассмотренная выше модульная архитектура ПО системы управления позволяет учесть особенности построения СЭУ на МСФ. Очевидно, что СЭУ различных классов будут отличаться друг от друга наличием или отсутствием ряда программных модулей, например, для функционирования СЭУ первого класса достаточно наличия трех верхних программных модулей, показанных на рисунке, для систем второго класса, помимо этого, необходимо наличие модуля имитационного моделирования и т.д. Вместе с тем предложенное структурирование системы на подсистемы и модули достаточно универсально и инвариантно к различным объектам управления. При этом, несмотря на общее функциональное назначение рассмотренных программных модулей для систем различных классов, непосредственная их программная реализация может существенно различаться в зависимости от сложности конкретного объекта управления, для которого проектируется СЭУ.

Предлагается методика построения архитектуры ПО СЭУ на МСФ, состоящая из следующих основных этапов:

- анализ режимов функционирования объекта управления;
- выявление основных факторов, влияющих на изменение состояния функционирования;
- определение возможных значений компонентов переменной состояния функционирования;
- формирование МСФ с выделением подмножеств наиболее вероятных и критических состояний функционирования;
- определение класса проектируемой СЭУ на МСФ;
- формирование состава ПО СЭУ (подсистем и модулей);
- определение связей подсистем и модулей системы друг с другом.

В дальнейшем на основе разработанной архитектуры осуществляется непосредственная программная реализация всех подсистем и модулей, входящих в состав ПО СЭУ. При этом возможно использование как специализированного инструментального ПО, например SCADA-систем, так и программных средств разработки приложений на базе CASE- и RAD-технологий. В частности, первый подход очень удобен при разработке СЭУ на базе существующей системы управления производством и позволяет, не создавая систему с нуля, добавить новые модули или расширить функциональность уже имеющихся в системе модулей.

### Заключение

В статье рассмотрена методика построения программной архитектуры СЭУ многомерными технологическими объектами. Решена задача структурирования ПО СЭУ с выделением функциональных подсистем и модулей. Предложена общая архитектура ПО СЭУ на МСФ, позволяющая разрабатывать системы управления различных классов.

Рассмотренная методика применялась при разработке программной архитектуры СЭУ динамическими режимами сушильных установок вальцеленточного и барабанного типов. Разработанные системы управления [10] позволили снизить затраты энергоресурсов в динамических режимах работы на 5–10 % и обеспечить требуемые значения качественных показателей выпускаемой продукции.

*Работа выполнена при финансовой поддержке гранта РФФИ, проект №17-08-00457-а.*

### Литература

1. Goodwin G.C., Graebe S.F., Salgado M.E. Control system design. NJ, Prentice Hall, 2000, 908 p.
2. Khalil H.K. Adaptive output feedback control of nonlinear systems represented by input-output models. Proc. IEEE Trans. Automat. Control, 1996, vol. 41, no. 2, pp. 177–188.
3. Муромцев Ю.Л., Ляпин Л.Н., Попова О.В. Моделирование и оптимизация сложных систем при изменениях состояния функционирования. Воронеж: Изд-во ВГУ, 1992. 164 с.
4. Муромцев Ю.Л., Тютюнник В.М. Включаемость в моделирование информационных систем // Информационные системы и процессы: сб. науч. тр. 2006. Вып. 4. С. 5–17.
5. Блохин А.Н. Моделирование развивающихся систем на множестве состояний функционирования // Вестн. Тамбов. гос. технич. ун-та. 2009. Т. 15. № 1. С. 17–20.
6. Еремин Е.Л. Адаптивное управление динамическим объектом на множестве состояний функционирования // Информатика и системы управления. 2012. № 4. С. 107–118.
7. Артемова С.В. Информационная система оптимального управления тепло-технологическими аппаратами. М.: МИНЦ, 2011. 234 с.
8. Чернышов Н.Г., Белоусов О.А. Система управления удаленными динамическими объектами с использованием принципов когнитивной графики // Автоматизация и современные технологии. 2010. № 7. С. 28–33.
9. Грибков А.Н., Муромцев Д.Ю., Шамкин В.Н., Белоусов О.А., Беляев М.П. Архитектура программного обеспечения систем энергосберегающего управления и контроля технологи-

ческих параметров многомерных объектов на множестве состояний функционирования // Проблемы машиноведения: матер. II Междунар. науч.-технич. конф. Омск: Изд-во Омского гос. техн. ун-та. 2018. С. 190–194.

10. Грибков А.Н., Муромцев Д.Ю. Информационно-управляющие системы многомерными технологическими объектами: теория и практика: монография. Тамбов: Изд-во ТГТУ, 2016. 164 с.

Software & Systems  
DOI: 10.15827/0236-235X.124.786-790

Received 06.07.18  
2018, vol; 31, no; 4, pp; 786–790

### A methodology for constructing software architecture of energy efficient management plans for MIMO process facilities on a set of functioning states

A.N. Gribkov<sup>1</sup>, Dr.Sc. (Engineering), Associate Professor, GribkovAlexey@yandex.ru

D.Yu. Muromtsev<sup>1</sup>, Dr.Sc. (Engineering), Professor, mdjur@mail.ru

V.N. Shamkin<sup>1</sup>, Dr.Sc. (Engineering), Associate Professor, shamkin-v@mail.ru

I.V. Tyurin<sup>1</sup>, Ph.D. (Engineering), Associate Professor, tyrinilja@yandex.ru

<sup>1</sup> Tambov State Technical University, Sovetskaya St. 106, Tambov, 392000, Russian Federation

**Abstract.** The paper considers theoretical and practical aspects of constructing a software architecture of energy efficient management plans for MIMO process facilities on a set of functioning states. The classification of control systems takes into account changes in functioning states during operation. The authors describe the possibilities of identifying the current functioning state in a control time slot. They also structure the software of energy efficient management plans with allocating functional subsystems and software modules.

The general architecture of a energy efficient management plan includes a knowledge and data management subsystem, an interface subsystem and six basic software modules. The knowledge and data management subsystem includes a knowledge base, an inference engine, a database and a database management system. The interface subsystem consists of an initial data input module, a cognitive graphics module and an integrated development environment.

The basic software modules of the energy efficient management plan are the following: an identification module of a control object dynamics model, a module for analysis of optimal control tasks, a module for synthesis of optimal control actions, a simulation module, a module for identification of a current functioning state, and an experiment planning module. Each basic module has determined system classes on a set of functioning states, in the software architecture of which it can be included.

The paper proposes a methodology for constructing a software architecture of different energy efficient management plans on a set of functioning states. It also considers the features of software implementation of energy efficient management plans based on various approaches (using applied and software tools).

**Keywords:** software, software architecture, energy efficient management plan, MIMO process facility, set of functioning states.

**Acknowledgements.** The study has been financially supported by a grant from the Russian Foundation for Basic Research, project no. 17-08-00457-a.

### References

1. Goodwin G.C., Graebe S.F., Salgado M.E. *Control System Design*. New Jersey, Prentice Hall Publ., 2000, 908 p.
2. Khalil H.K. Adaptive output feedtach control of nonlinear systems represented by input-output models. *IEEE Trans. Automat. Control*. 1996, vol. 41, no. 2, pp. 177–188.
3. Muromtsev Yu.L., Lyapin L.N., Popova O.V. *Modeling and Optimization of Complex Systems with Operation State Changes*. Voronezh, VGU Publ., 1992, 164 p.
4. Muromtsev Yu.L., Tyutyunnik V.M. Inclusiveness in information system modeling. *Information Systems and Processes: Proc.* 2006, iss. 4, pp. 5–17 (in Russ.).
5. Blokhin A.N. Evolutionary system modeling on state set of functioning. *Trans. of TSTU*. 2009, vol. 15, no. 1, pp. 17–20 (in Russ.).
6. Eremin E.L. Adaptive control of a dynamic object on the set of states of functioning. *Information Science and Control Systems*. 2012, no. 4, pp. 107–118 (in Russ.).
7. Artemova S.V. *An Information System of Optimal Control of Heat-Technological Devices*. Monograph. Moscow, St. Petersburg, Vena, Gamburg, MINTs Publ., 2011, 234 p.
8. Chernyshov N.G., Belousov O.A. Remote dynamic objects control system using the principles of cognitive graphics. *Automation and Modern Technologies*. 2010, no. 7, pp. 28–33 (in Russ.).
9. Gribkov A.N., Muromtsev D.Yu., Shamkin V.N., Belousov O.A., Belyaev M.P. Software architecture for energy efficient management and control plans for process parameters of MIMO systems on a set of functioning states. *Engineering Problems: Proc. 2nd Intern. Sci. and Tech. Conf.* Omsk, 2018, pp. 190–194 (in Russ.).
10. Gribkov A.N., Muromtsev D.Yu. *Information and Control Systems of MIMO Process Facilities: Theory and Practice*. Monograph. Tambov, TGTU Publ., 2016, 164 p.