

МЕХАТРОНИКА, АВТОМАТИЗАЦИЯ, УПРАВЛЕНИЕ

Том 17

2016

№ 11

Издается с 2000 года

ISSN 1684-6427

DOI 10.17587/issn.1684-6427

СОДЕРЖАНИЕ

МЕТОДЫ ТЕОРИИ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ

Шэнь Кай, Неусыпин К. А. Исследование критериев степеней наблюдаемости, управляемости и идентифицируемости линейных динамических систем 723

Васильев Е. М., Одношвикин А. С. Гомеостатическое управление динамическими системами на основе нечетких регуляторов 732

РОБОТОТЕХНИЧЕСКИЕ СИСТЕМЫ

Каляев И. А., Каляев А. И., Коровин Я. С. Принципы организации и функционирования безлюдного роботизированного производства с децентрализованным диспетчером 741

Филимонов А. Б., Филимонов Н. Б., Тихонов В. Ю. Задача прохождения лабиринта интеллектуальными агентами 750

Страшнов Е. В., Торгашев М. А. Моделирование динамики электроприводов виртуальных роботов в имитационно-тренажерных комплексах 762

Горитов А. Н. Алгоритм синтеза промышленных роботов на основе заданного набора модулей 769

УПРАВЛЕНИЕ В АВИАКОСМИЧЕСКИХ СИСТЕМАХ

Зенкевич С. Л., Галустьян Н. К. Децентрализованное управление группой квадрокоптеров 774

Лебедев Г. Н., Гончаренко В. И., Румакина А. В. Модификация метода ветвей и границ для двумерной маршрутизации координированного полета группы летательных аппаратов 783

Редакционный совет:

CHYI-YEU LIN, PhD, prof.
GROUMPOS P. P., prof.
JEN-HWA GUO, PhD, prof.
KATALINIC B., PhD, prof.
SUBUDHI B., PhD, prof.
АЛИЕВ Т. А., акад. НАНА, проф.
АНШАКОВ Г. П., чл.-корр. РАН, проф.
БОЛОТНИК Н. Н., чл.-корр. РАН, проф.
ВАСИЛЬЕВ С. Н., акад. РАН, проф.
КАЛЯЕВ И. А., чл.-корр. РАН, проф.
КРАСНЕВСКИЙ Л. Г., чл.-корр. НАНБ, проф.
КУЗНЕЦОВ Н. А., акад. РАН, проф.
ЛЕОНОВ Г. А., чл.-корр. РАН, проф.
МАТВЕЕНКО А. М., акад. РАН, проф.
МИКРИН Е. А., акад. РАН, проф.
ПЕШЕХОНОВ В. Г., акад. РАН, проф.
РЕЗЧИКОВ А. Ф., чл.-корр. РАН, проф.
СЕБРЯКОВ Г. Г., чл.-корр. РАН, проф.
СИГОВ А. С., акад. РАН, проф.
СОЙФЕР В. А., чл.-корр. РАН, проф.
СОЛОВЬЕВ В. А., чл.-корр. РАН, проф.
СОЛОМЕНЦЕВ Ю. М., чл.-корр. РАН, проф.
ФЕДОРОВ И. Б., акад. РАН, проф.
ЧЕНЦОВ А. Г., чл.-корр. РАН, проф.
ЧЕРНОУСЬКО Ф. Л., акад. РАН, проф.
ШЕРБАТЮК А. Ф., чл.-корр. РАН, проф.
ЮСУПОВ Р. М., чл.-корр. РАН, проф.

Главный редактор:

ФИЛИМОНОВ Н. Б., д. т. н., с. н. с.

Заместители гл. редактора:

ПОДУРАЕВ Ю. В., д. т. н., проф.
ПУТОВ В. В., д. т. н., проф.
ЮЩЕНКО А. С., д. т. н., проф.

Ответственный секретарь:

БЕЗМЕНОВА М. Ю.

Редакционная коллегия:

АЛЕКСАНДРОВ В. В., д. ф.-м. н., проф.
АНТОНОВ Б. И.
АРШАНСКИЙ М. М., д. т. н., проф.
БУКОВ В. Н., д. т. н., проф.
ВИТТИХ В. А., д. т. н., проф.
ГРАДЕЦКИЙ В. Г., д. т. н., проф.
ЕРМОЛОВ И. Л., д. т. н., доц.
ИВЧЕНКО В. Д., д. т. н., проф.
ИЛЬЯСОВ Б. Г., д. т. н., проф.
КОЛОСОВ О. С., д. т. н., проф.
КОРОСТЕЛЕВ В. Ф., д. т. н., проф.
ЛЕБЕДЕВ Г. Н., д. т. н., проф.
ЛОХИН В. М., д. т. н., проф.
ПАВЛОВСКИЙ В. Е., д. ф.-м. н., проф.
ПРОХОРОВ Н. Л., д. т. н., проф.
ПШИХОПОВ В. Х., д. т. н., проф.
РАПОПОРТ Э. Я., д. т. н., проф.
СЕРГЕЕВ С. Ф., д. пс. н., с. н. с.
ФИЛАРЕТОВ В. Ф., д. т. н., проф.
ФРАДКОВ А. Л., д. т. н., проф.
ФУРСОВ В. А., д. т. н., проф.
ЮРЕВИЧ Е. И., д. т. н., проф.

Редакция:

ГРИГОРИН-РЯБОВА Е. В.

Журнал входит в Перечень рецензируемых научных изданий, в которых должны быть опубликованы основные результаты диссертаций на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук; журнал включен в систему Российского индекса научного цитирования

Информация о журнале доступна по сети Internet по адресу:
<http://novtex.ru/mech>, e-mail: mech@novtex.ru

MECHATRONICS, AUTOMATION, CONTROL



МЕХАТРОНИКА, АВТОМАТИЗАЦИЯ, УПРАВЛЕНИЕ

Published since 2000

ISSN 1684-6427

DOI 10.17587/issn.1684-6427

Editorial Council:

ALIEV T. A., prof., Azerbaijan, Baku
 ANSHAKOV G. P., Russia, Samara
 BOLOTNIK N. N., Russia, Moscow
 CHENTSOV A. G., Russia, Ekaterinburg
 CHERNOUSKO F. L., Russia, Moscow
 CHYI-YEU LIN, PhD, Prof., Taiwan, Taipei
 FEDOROV I. B., Russia, Moscow
 GROUMPOS P. P., prof., Greece, Patras
 JEN-HWA GUO, PhD, Prof., Taiwan, Taipei
 KALYAEV I. A., Russia, Taganrog
 KATALINIC B., PhD, Prof., Austria, Vienna
 KRASNEVSKIY L. G., Belarus, Minsk
 KUZNETSOV N. A., Russia, Moscow
 LEONOV G. A., Russia, S.-Peterburg
 MATVEENKO A. M., Russia, Moscow
 MIKRIN E. A., Russia, Moscow
 PESHEKHONOV V. G., Russia, S.-Peterburg
 REZCHIKOV A. F., Russia, Saratov
 SCHERBATYUK A. F., Russia, Vladivostok
 SEBRYAKOV G. G., Russia, Moscow
 SIGOV A. S., Russia, Moscow
 SOJFER V. A., Russia, Samara
 SOLOMENTSEV Yu. M., Russia, Moscow
 SOLOVJEV V. A., Russia, Moscow
 SUBUDHI B., PhD, Prof., India, Sundargarh
 VASILYEV S.N., Russia, Moscow
 YUSUPOV R. M., Russia, S.-Peterburg

Editor-in-Chief:

FILIMONOV N. B., Russia, Moscow

Deputy Editor-in-Chief:

PODURAEV Yu. V., Russia, Moscow
 PUTOV V. V., Russia, S.-Peterburg
 YUSCHENKO A. S., Russia, Moscow

Responsible Secretary:

BEZMEANOVA M. Yu., Russia, Moscow

Editorial Board:

ALEXANDROV V. V., Russia, Moscow
 ANTONOV B. I., Russia, Moscow
 ARSHANSKY M. M., Russia, Tver
 BUKOV V. N., Russia, Zhukovsky
 ERMOLOV I. L., Russia, Moscow
 FILARETOV V. F., Russia, Vladivostok
 FRADKOV A. L., Russia, S.-Peterburg
 FURSOV V. A., Russia, Samara
 GRADETSKY V. G., Russia, Moscow
 ILYASOV B. G., Russia, Ufa
 IVCHENKO V. D., Russia, Moscow
 KOLOSOV O. S., Russia, Moscow
 KOROSTELEV V. F., Russia, Vladimir
 LEBEDEV G. N., Russia, Moscow
 LOKHIN V. M., Russia, Moscow
 PAVLOVSKY V. E., Russia, Moscow
 PROKHOROV N. L., Russia, Moscow
 PSHIKHOPOV V. Kh., Russia, S.-Peterburg
 RAPOPORT E. Ya., Russia, Samara
 SERGEEV S. F., Russia, S.-Peterburg
 VITTIKH V. A., Russia, Samara
 YUREVICH E. I., Russia, S.-Peterburg

Editorial Staff:

GRIGORIN-RYABOVA E.V., Russia, Moscow

The mission of the Journal is to cover the current state, trends and prospective development of *mechatronics*, that is the priority field in the technosphere as it combines mechanics, electronics, automatics and informatics in order to improve manufacturing processes and to develop new generations of equipment. Covers topical issues of development, creation, implementation and operation of mechatronic systems and technologies in the production sector, power economy and in transport.

CONTENTS

METHODS OF THE THEORY OF AUTOMATIC CONTROL

- Shen Kai, Neusypin K. A.** Study of the Criteria for the Degrees of Observability, Controllability and Identifiability of the Linear Dynamical Systems 723
- Vasiljev E. M., Odnoshivkin A. S.** Homeostatic Control of the Dynamic Systems on the Basis of the Fuzzy Controllers 732

ROBOTIC SYSTEMS

- Kalyaev I. A., Kalyaev A. I., Korovin I. S.** Principles of Organization and Operation of an Unmanned Robotic Production with a Decentralized Controller. 741
- Filimonov A. B., Filimonov N. B., Tikhonov V. Ju.** The Problem of Maze Passage by Intelligent Agents 750
- Strashnov E. V., Torgashev M. A.** Simulation of the Actuator Dynamics of the Virtual Robots in the Training Complexes 762
- Goritov A. N.** Synthesis Algorithm for Industrial Robots Based on a Given Set of Modules . . . 769

CONTROL IN AEROSPACE SYSTEMS

- Zenkevich S. L., Galustyan N. K.** Decentralized Control of a Quadrocopter Swarm 774
- Lebedev G. N., Goncharenko V. I., Roumakina A. V.** Modification of the Integer Branch and Bound Method for Solving of a Two-Dimensional Routing Task of an Aircraft Group's Coordinated Flight. 783

Information about the journal is available online at:
<http://novtex.ru/mech.html>, e-mail: mech@novtex.ru

Кай Шэнь, аспирант, shenkaichn@mail.ru,
Нанкинский университет науки и технологий,
К. А. Неусыпин, д-р техн. наук, проф., neysipin@mail.ru,
МГТУ им. Н. Э. Баумана

Исследование критериев степеней наблюдаемости, управляемости и идентифицируемости линейных динамических систем¹

Рассмотрены подходы к решению задач определения степени наблюдаемости, управляемости и идентифицируемости параметров линейных динамических систем. Исследованы численные критерии вычисления степени наблюдаемости, управляемости компонент вектора состояния и степени идентифицируемости конкретных параметров модели динамического объекта с использованием скалярного подхода. Указаны примеры практического использования исследуемых критериев и намечены дальнейшие направления исследований.

Ключевые слова: модель динамического объекта, степень идентифицируемости, степень управляемости, степень наблюдаемости

Введение

Решение задач управления различными динамическими объектами предполагает использование математической модели исследуемого процесса. Изменение параметров динамической системы в процессе функционирования, а также изменение параметров и/или структуры математической модели приводит к изменению статистических и динамических свойств исследуемой системы.

В теории управления для определения свойств систем используются такие понятия, как устойчивость, наблюдаемость, управляемость, идентифицируемость. Известны разнообразные критерии оценки этих свойств [1–3]. Однако в практических приложениях часто недостаточно получить принципиальный ответ на вопрос: устойчива, наблюдаема, управляема, идентифицируема система или нет? Желательно оценить качества исследуемой динамической системы: максимум ошибки, быстродействие, различные интегральные оценки, запас устойчивости, чувствительность, степени наблюдаемости, управляемости и идентифицируемости [4–7].

Запас устойчивости определяется путем решения дифференциального уравнения замкнутой системы, подверженной влиянию внешних возмущений. Другой подход предполагает исследование значений запасов устойчивости по амплитуде и по фазе, которые определяются по годографу комплексной частотной характеристики разомкнутого контура и

логарифмическим амплитудно- и фазово-частотным характеристикам [1].

Весьма распространенным показателем качества системы с обратной связью является функция чувствительности (ФЧ). Под чувствительностью понимают зависимость характеристик исследуемой системы от изменения ее параметров. ФЧ — это частные производные от координат системы или показателей качества процессов по вариациям параметров [6–8]. Чем больше чувствительность, тем сильнее влияет исследуемый параметр на выходной сигнал системы.

М. Л. Быховский разработал структурный метод построения моделей для вычисления ФЧ [8]. В практических приложениях теория чувствительности, основанная на понятии ФЧ, нашла различные приложения, например, в задачах синтеза систем управления [9]; для совершенствования моделей с применением ФЧ и схемы замещения системы передачи и распределения электроэнергии [10]; в методе определения дальности цели пассивными комплексами [11].

Критерии оценки качества системы с помощью запаса устойчивости и ФЧ хорошо теоретически отработаны и имеют широкое практическое применение. Данные качественные характеристики моделей системы имеют явно выраженную связь: мерой чувствительности системы к параметрическим возмущениям служат запасы устойчивости по амплитуде и фазе.

Другие качественные характеристики моделей динамических систем — показатели степени наблюдаемости, управляемости и идентифицируемости — разработаны не так подробно, как запас устойчи-

¹ Работа выполнена при поддержке the Programme of Introducing Talents of Discipline to Universities in P. R. China ("программа 111", No. B 16025).

ности и чувствительность, представлены отдельными критериями, и создание общей теории является перспективной задачей.

Известные критерии [1—4] определения степени наблюдаемости и степени управляемости позволяют определить лишь то, какие из компонент одного вектора состояния наблюдаются или управляются лучше. Эти критерии дают только относительную оценку качественных характеристик компонент конкретного вектора состояния исследуемой системы и не позволяют проводить сравнение компонент векторов состояния различных систем. Поэтому они неудобны для использования при сравнении качества наблюдения, управления и идентификации в общем случае. Обычно в практических приложениях необходимо знать, возможно ли эффективное наблюдение и управление каждой конкретной компонентой вектора состояния. Для этого введено понятие меры или степени наблюдаемости (управляемости) [12] каждой конкретной переменной состояния. При проведении параметрической идентификации также целесообразно знать качественные характеристики этого процесса, которые определяются степенью идентифицируемости каждого исследуемого параметра матрицы модели.

Вопрос о степени наблюдаемости ("не только наблюдаемы, а как наблюдаемы") впервые рассмотрен Р. Г. Брауном в 1966 г. [13]. После этого было предложено несколько критериев степени наблюдаемости. Сначала Х. Л. Аблином определен критерий степени наблюдаемости с помощью взаимного отношения значений ошибок оценивания переменных вектора состояния и ошибок наблюдения (измерения) [14]. Затем Ф. М. Хамом и Р. Г. Брауном доказано, что собственные числа и собственные векторы ковариационной матрицы ошибок оценивания могут предоставить полезную информацию о наблюдаемости системы [15]. С точки зрения точности оценивания степень наблюдаемости может определяться соотношением дисперсии произвольной компоненты вектора состояния и дисперсии непосредственно измеряемого вектора состояния [16, 17].

В настоящей статье авторы сделали важный шаг к обобщению результатов в этой области. Материал содержит результаты, касающиеся линейных динамических систем. Известные результаты в исследуемой области для нелинейных систем носят фрагментарный характер и по этой причине не рассмотрены. Однако следует отметить такие многообещающие направления исследований и полученные результаты, как применение современной дифференциальной геометрии для переформулирования в геометрических терминах понятий наблюдаемости и управляемости для нелинейных систем, которые проводятся А. Г. Кушнером.

Критерии наблюдаемости, управляемости и идентифицируемости

Наблюдаемость, управляемость и идентифицируемость играют важную роль при синтезе систем управления динамическими объектами, оценивании состояния и идентификации их параметров.

Понятия наблюдаемости и управляемости дуальны, т.е. если система полностью наблюдаема, то построенная для этой системы сопряженная система будет полностью управляема. Справедливо и обратное утверждение. Следовательно, для определения наблюдаемости можно построить сопряженную систему, которую затем исследовать с помощью какого-либо критерия управляемости, что существенно расширяет методологический аппарат для исследования этих характеристик.

Заметим, что критерий полной управляемости не связан как-либо с устойчивостью системы. Поэтому неустойчивая система может быть полностью управляемой и наоборот. Полная управляемость означает стабилизируемость системы, т.е. возможность путем присоединения регулятора создать замкнутую систему с желаемым распределением собственных значений.

Одним из популярных критериев является критерий Калмана [1—3], который отличается простотой и широко используется в практических приложениях.

Пусть объект описывается уравнениями вида

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{z}(t) &= \mathbf{C}\mathbf{x}(t), \end{aligned} \quad (1)$$

где \mathbf{x} — n -вектор состояния; \mathbf{u} — m -вектор управления; \mathbf{z} — l -вектор измерений; \mathbf{A} — $(n \times n)$ -матрица системы; \mathbf{B} — $(n \times m)$ -матрица управления; \mathbf{C} — $(l \times n)$ -матрица измерений.

Система (1) называется *полностью наблюдаемой* на интервале времени $[0, t_1]$, если вектор состояния $\mathbf{x}(t_0)$ можно определить по известному вектору измерений $\mathbf{z}(t)$. Проверку наблюдаемости можно осуществить, воспользовавшись критерием полной наблюдаемости Калмана. Система (1) является полностью наблюдаемой, если ранг матрицы наблюдаемости \mathbf{S} равен порядку системы n , т.е. если измерения $\mathbf{z}(t)$ содержат достаточную информацию для определения $\mathbf{x}(t_0)$. Матрица наблюдаемости \mathbf{S} имеет вид

$$\mathbf{S} = [\mathbf{C}^T | \mathbf{A}^T \mathbf{C}^T | (\mathbf{A}^T)^2 \mathbf{C}^T | \dots | (\mathbf{A}^T)^{n-1} \mathbf{C}^T]. \quad (2)$$

В случае если ранг матрицы наблюдаемости меньше порядка системы, то по измерениям $\mathbf{z}(t)$ можно оценить лишь часть вектора состояния $\mathbf{x}(t)$.

Другой критерий предусматривает каноническое преобразование, в результате которого получается система уравнений с некратными собственными значениями, связь между которыми по переменным состояниям отсутствует [4]. В этом случае критерий полной управляемости системы заключается в отсутствии нулевых элементов в ка-

нонической матрице системы $\mathbf{B}^* = [b_{ij}^*]$ [4]. Если в матрице \mathbf{B}^* все элементы i -й строки равны нулю, то любым выбором управления u_j нельзя изменить переменную x_i , т.е. система не является полностью управляемой. Напротив, если в матрице \mathbf{B}^* нет строк с нулевыми элементами, то можно любую переменную состояния x_i^* перевести от значения $x_i^*(t_0)$ к любому желаемому значению $x_i^*(t_1)$, т.е. система полностью управляема. Это можно видеть из решения системы:

$$x_i^*(t_1) = e^{\lambda_i(t_1-t_0)} x_i^*(t_0) + \sum_{j=1}^m \int_{t_0}^{t_1} e^{\lambda_i(t_1-\tau)} b_{ij}^* u_j(\tau) d\tau. \quad (3)$$

Если строки матрицы \mathbf{B}^* линейно зависимы друг друга, то система тоже не полностью управляема.

Известны различные критерии определения степени идентифицируемости или условия определенности [5–7].

Критерий, предложенный Н. А. Балониным [5], позволяет определить принципиальную возможность осуществления процедуры идентификации.

Модель линейной однородной системы имеет вид

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t), \quad (4)$$

где вектор состояния $\mathbf{x} \in R^n$, $\mathbf{x}_0 = \mathbf{x}(0)$.

Линейная однородная система называется *полностью идентифицируемой по вектору состояния*, если при заданном векторе начальных условий \mathbf{x}_0 матрица параметров \mathbf{A} может быть однозначно восстановлена за конечный отрезок времени идентификации по одной временной последовательности $\mathbf{x} = \mathbf{x}(t)$.

Иначе, пара $(\mathbf{A}, \mathbf{x}_0)$ *полностью идентифицируема или идентифицируема вполне*, когда множество пар $(\mathbf{A}, \mathbf{x}_0)$, объединенных общностью интегральной кривой $\mathbf{x} = \mathbf{x}(t)$, $\mathbf{x}_0 = \mathbf{x}(0)$, вырождается в точку \mathbf{A} . В противном случае пара неидентифицируема.

Необходимое и достаточное условие полной идентифицируемости пары $(\mathbf{A}, \mathbf{x}_0)$ имеет следующий вид:

$$\text{ранг}[\mathbf{W}_0] = \text{ранг}[\mathbf{x}_0, \mathbf{A}\mathbf{x}_0, \mathbf{A}^2\mathbf{x}_0, \dots, \mathbf{A}^{n-1}\mathbf{x}_0] = n, \quad (5)$$

где \mathbf{W}_0 — матрица идентифицируемости.

Данный критерий предполагает определение фундаментальной возможности идентификации параметров динамической системы.

Известен критерий, предложенный А. В. Балакришнаном [6], основанный на конкретном критерии идентифицируемости.

Анализируется линейная модель сигнала с известной и наблюдаемой дисперсией. Исследован процесс оценивания матрицы \mathbf{B} по наблюдаемым последовательно данным, а не по всей выборке, при известном векторе управления \mathbf{u} и предположении, что матрица \mathbf{A} устойчива.

Для получения оценки безусловного максимума правдоподобия нужно использовать все имеющиеся данные, и, кроме того, дополнительная трудность

состоит в необходимости оценивания начального состояния. При построении оценки, которая является последовательной, или "текущей", и не использует какую-либо оценку состояния, необходимо предположить, что выполнено "условие определенности", которое накладывает ограничение на входную последовательность, т.е. при выполнении условия определенности (и устойчивости матрицы \mathbf{A}) ковариационная матрица погрешности

$$E[(\mathbf{B} - \hat{\mathbf{B}}_n)(\mathbf{B} - \hat{\mathbf{B}}_n)^*] \rightarrow 0 \text{ при } n \rightarrow \infty. \quad (6)$$

Вычисление критерия идентифицируемости по данной методике предполагает сложные математические вычисления, поэтому в практическом плане применять его затруднительно.

Другой известный критерий идентифицируемости предложен С. А. Айвазяном [7]. Исследуется система одновременных уравнений в структурной форме, определяющих связь между экзогенными и эндогенными переменными.

Рассмотрим некоторые наиболее важные типы ограничений и приведем критерии идентифицируемости. Предполагается, что априорные ограничения являются линейными однородными функциями, каждая из которых зависит только от коэффициентов матрицы \mathbf{Q} , которые могут быть однозначно восстановлены по матрице приведенной формы $\mathbf{\Pi} = -\mathbf{D}^{-1}\mathbf{\Gamma}$, где \mathbf{D} — матрица, состоящая из коэффициентов при эндогенных переменных; $\mathbf{\Gamma}$ — матрица, состоящая из коэффициентов предопределенных переменных.

Пусть \mathbf{I}_k — $(k \times k)$ -единичная матрица. Введем

$$\text{обозначение } \mathbf{W} = \begin{bmatrix} \mathbf{\Pi} \\ \mathbf{I}_k \end{bmatrix}.$$

Соотношение $\mathbf{D}\mathbf{\Pi} + \mathbf{\Gamma} = 0$ между структурной и приведенной формами может быть записано в виде

$$\mathbf{Q}\mathbf{W} = 0, \quad (7)$$

где \mathbf{Q} — матрица порядка $g \times (g + k)$; \mathbf{W} — матрица порядка $(g + k) \times k$, имеющая ранг k .

Пусть $\alpha_1 \mathbf{F} = 0$, где \mathbf{F} — матрица из $g + k$ строк, имеющая столько столбцов, сколько ограничений; α_1 — первая строка матрицы \mathbf{Q} . Например, пусть априори известно, что коэффициент системы одновременных уравнений в структурной форме $\beta_{12} = 0$. Тогда один из столбцов матрицы \mathbf{F} имеет вид $(0, 1, 0, \dots, 0)^T$.

Элементы вектора α_1 являются решением системы уравнений

$$\alpha_1[\mathbf{W}\mathbf{F}] = 0. \quad (8)$$

В силу правила нормализации для идентифицируемости первого уравнения (8) необходимо и достаточно, чтобы пространство решений системы было одномерным, т.е.

$$\text{rg}[\mathbf{W}\mathbf{F}] = g + k - 1. \quad (9)$$

Пусть η — число ограничений. Тогда матрица $[\mathbf{WF}]$ содержит $k + \eta$ столбцов и для выполнения (9) необходимо, чтобы $\eta \geq g - 1$, т.е. для идентифицируемости какого-либо из уравнений необходимо, чтобы число ограничений было не меньше числа уравнений модели, уменьшенного на единицу.

Если имеются только исключаяющие ограничения, т.е. априорная информация о равенстве нулю некоторых коэффициентов, то необходимое условие идентифицируемости определенного уравнения заключается в следующем: число исключенных из уравнения экзогенных переменных должно быть не меньше числа участвующих в нем эндогенных переменных, уменьшенного на единицу.

Сформулированные необходимые условия (так называемые правила порядка) в силу своей простоты являются весьма полезными при решении проблемы идентифицируемости, поскольку при построении модели они позволяют сразу выявить неидентифицируемые уравнения. Однако эти условия могут оказаться далекими от достаточных. Необходимое и достаточное условие (9) не подходит для проверки идентифицируемости модели, поскольку требует построения матрицы $\mathbf{П}$. На основе соотношения (9) сформулирован критерий идентифицируемости.

Первое уравнение системы идентифицируемо тогда и только тогда, когда $\text{rg}[\mathbf{QF}] = g - 1$. Это утверждение можно получить из соображений, связанных с инвариантностью коэффициентов при умножении структурной формы на допустимые матрицы. Этот метод обычно применяется только для теоретических абстрактных задач.

На практике часто применяется критерий идентифицируемости в алгебраической форме, полученный на основе метода матричных делителей нуля [18, 19].

Представленные критерии идентифицируемости не позволяют проводить сравнение качества идентификации параметров различных моделей и не всегда удобны в применении на практике.

Таким образом, все упомянутые критерии определения степеней наблюдаемости, управляемости и идентифицируемости неудобны для использования при сравнении качественных характеристик в общем случае.

Критерии степени наблюдаемости, управляемости и идентифицируемости

Критерии степени наблюдаемости. В работе [12] представлен критерий степени наблюдаемости, который позволяет выделить слабонаблюдаемые компоненты вектора состояния и сформировать эффективно оцениваемый вектор состояния. Другой известный критерий, позволяющий определить качество оценивания переменных состояния, предполагает проведение предварительных преобразований, включающих три этапа [15]: вычисление ковариационной матрицы ошибок оценивания;

нормализацию ковариационной матрицы ошибок оценивания; вычисление собственных чисел нормализованной ковариационной матрицы ошибок оценивания. Критерий степени наблюдаемости формулируется следующим образом: чем меньше собственное число, тем лучше наблюдаема компонента вектора состояния. Представленный критерий чрезвычайно неудобен в практическом применении, так как требует проведения большого объема предварительных вычислений.

В различных практических приложениях нашел широкое применение критерий степени наблюдаемости [16, 17], позволяющий определять степень наблюдаемости в виде скалярной величины. Рассмотрим этот критерий подробнее.

Пусть объект описывается уравнением вида [20]

$$\mathbf{x}_k = \mathbf{\Phi}\mathbf{x}_{k-1} + \mathbf{G}\mathbf{w}_{k-1}, \quad (10)$$

где \mathbf{x}_{k-1} — n -вектор состояния; \mathbf{w}_{k-1} — l -вектор входного шума, который является дискретным аналогом белого гауссового шума с нулевым математическим ожиданием; $\mathbf{\Phi}$ — $(n \times n)$ -матрица системы; \mathbf{G} — $(n \times l)$ -матрица входного шума.

Часть вектора состояния измеряется:

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k, \quad (11)$$

где \mathbf{z}_k — m -вектор измерений; \mathbf{H} — $(m \times n)$ -матрица измерений; \mathbf{v}_k — m -вектор измерительного шума, который является дискретным аналогом белого гауссового шума с нулевым математическим ожиданием, причем \mathbf{v} и \mathbf{w} некоррелированы между собой, т.е. $E[\mathbf{v}_j\mathbf{w}_k^T] = 0$.

Используем скалярный подход [16]: не теряя общности постановки задачи, предположим, что измеряется одна компонента вектора состояния, т.е. $\mathbf{H} = [1, 0, \dots, 0]$. Разобьем каждый шаг измерений на n подтактов и выразим эти измерения через вектор состояния на первом подтакте измерений:

$$\begin{aligned} z_1 &= \mathbf{H}\mathbf{x}_1 + v_1, \\ z_2 &= \mathbf{H}\mathbf{\Phi}\mathbf{x}_1 + \mathbf{H}\mathbf{w}_1 + v_2, \\ &\dots \dots \dots \end{aligned} \quad (12)$$

$$z_n = \mathbf{H}\mathbf{\Phi}^{n-1}\mathbf{x}_1 + \mathbf{H}\mathbf{\Phi}^{n-2}\mathbf{w}_1 + \dots + \mathbf{H}\mathbf{w}_{n-1} + v_n$$

или в матричной форме:

$$\mathbf{z}^* = \mathbf{S}\mathbf{x}_1 + \mathbf{v}^*, \quad (13)$$

$$\text{где } \mathbf{z}^* = \begin{bmatrix} z_1 \\ z_2 \\ \dots \\ z_n \end{bmatrix}, \mathbf{S} = \begin{bmatrix} \mathbf{H} \\ \mathbf{H}\mathbf{\Phi} \\ \dots \\ \mathbf{H}\mathbf{\Phi}^{n-1} \end{bmatrix},$$

$$\mathbf{v}^* = \begin{bmatrix} v_1 \\ \mathbf{H}\mathbf{w}_1 + v_2 \\ \dots \dots \dots \\ \mathbf{H}\mathbf{\Phi}^{n-2}\mathbf{w}_1 + \dots + \mathbf{H}\mathbf{w}_{n-1} + v_n \end{bmatrix}.$$

Выразим из уравнения объекта вектор состояния в первом подтакте измерения:

$$\mathbf{x}_1 = \mathbf{S}^{-1}\mathbf{z}^* - \mathbf{S}^{-1}\mathbf{v}^*. \quad (14)$$

Введем обозначение $\mathbf{y} = \mathbf{S}^{-1}\mathbf{z}^*$ и запишем уравнения в скалярном виде:

$$y^i = a_{i1}z_1 + a_{i2}z_2 + \dots + a_{in}z_n. \quad (15)$$

Здесь y^i — элемент вектора \mathbf{y} , a_i — это i -я строка матрицы \mathbf{S}^{-1} .

Для остальных компонент вектора состояния уравнения измерений формулируются в соответствии с уравнением (15).

Введем понятие приведенного измерительного шума [17]. Для произвольной компоненты вектора состояния приведенный измерительный шум в соответствии с уравнением (15) имеет вид

$$v^{*i} = a_{i1}v_1 + a_{i2}v_2 + \dots + a_{in}v_n. \quad (16)$$

Дисперсия приведенного к i -й компоненте измерительного шума определяется коэффициентами $a_{i1}, a_{i2}, \dots, a_{in}$, т.е.

$$r^{*i} = E[(v^{*i})^2] = [a_{i1}^2 + a_{i2}^2 + \dots + a_{in}^2]r, \quad (17)$$

где $r = E[v^2]$ — дисперсия исходного измерительного шума v .

Численный критерий степени наблюдаемости.

Судить о степени наблюдаемости можно по двум характеристикам — точности оценивания и времени сходимости. Критерий степени наблюдаемости имеет вид [16, 17]

$$\lambda^i = \frac{E[(x^i)^2]r}{E[(y^i)^2]r^{*i}}. \quad (18)$$

Здесь $E[(x^i)^2]$ — дисперсия произвольной i -й компоненты вектора состояния; $E[(y^i)^2]$ — дисперсия непосредственно измеряемого вектора состояния.

Данный критерий применялся для выбора наилучшей структуры навигационного комплекса ЛА [21] в задаче синтеза адаптивного регулятора инерциальной навигационной системы [22].

Критерии степени управляемости. Критерий управляемости Калмана позволяет определить, управляем ли весь вектор состояния системы. Критерий, предложенный в работе [23], позволяет определить, какие компоненты вектора состояния управляемы, и выделить компоненты, которыми можно управлять наилучшим образом. Исключив из вектора состояния неуправляемые компоненты, можно достичь полной управляемости исследуемой системы.

Рассматривается в данном случае сопряженная система (19), и к ней применяется критерий полной управляемости:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= -\mathbf{A}^T\mathbf{x}_c(t) - \mathbf{C}^T\mathbf{u}_c(t); \\ \mathbf{z}_c(t) &= \mathbf{B}^T\mathbf{x}_c(t). \end{aligned} \quad (19)$$

Для того чтобы воспользоваться этим критерием, необходимо перейти от матричного уравнения системы к рассмотрению системы скалярных уравнений. Затем нужно перевести скалярные уравнения из пространства оригиналов в пространство изображений по Лапласу и разрешить полученную систему уравнений относительно каждой переменной. Анализуются найденные по правилу Крамера определители. Система вида полностью управляема, если найденные определители линейно независимы, и ни один из них не равен нулю.

Рассмотрим критерий управляемости вида

$$C_1\Delta_1 + C_2\Delta_2 + \dots + C_n\Delta_n \neq 0, \quad (20)$$

где Δ_i — правые части дифференциального уравнения рассматриваемой системы, т.е. решения этой системы относительно каждой переменной состояния; C_i — постоянные коэффициенты, которые выбираются произвольным образом, но хотя бы один из них не должен быть равным нулю.

Данный критерий позволяет получить непосредственную связь между компонентами вектора состояния и компонентами вектора управления. Эта связь позволяет выносить суждение о степени управляемости по той или иной переменной состояния. В случае одинаковой структуры выражений Δ по коэффициентам этих выражений можно судить о степени управляемости.

Степень управляемости можно определить, исследовав систему канонического вида, т.е. систему уравнений, в которой отсутствует связь по переменным состояния.

Запишем уравнение исследуемой системы в канонической форме [4]:

$$\dot{\mathbf{x}}_c^k(t) = \mathbf{A}^k\mathbf{x}_c^k(t) + \mathbf{C}^k\mathbf{u}_c^k(t), \quad (21)$$

здесь $\mathbf{A}^k = -\mathbf{V}^{-1}\mathbf{A}^T\mathbf{V}$, $\mathbf{C}^k = -\mathbf{V}^{-1}\mathbf{C}^T$, $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_n]^T$ — матрица канонического преобразования, \mathbf{v}_i — собственные векторы исследуемой системы.

Система является полностью управляемой, если матрица \mathbf{C}^k канонической системы (21) не содержит строк, все элементы которых равны нулю. Следует отметить, что данный критерий можно применять лишь в случае отсутствия кратных собственных значений системы. Степень управляемости характеризуют модули элементов строк матрицы \mathbf{C}^k . Представленный критерий позволяет проводить сравнительный анализ управляемости и определять, лучше или хуже управляемы компоненты вектора состояния по сравнению друг с другом. Лучше уп-

равляемы те компоненты вектора состояния, у которых модули элементов строк матрицы больше модулей соответствующих элементов у других строк.

Другим подходом для определения степени управляемости динамических систем является их исследование на динамических стендах и построение диаграмм, определяющих связь эффективности управления в различных режимах [24, 25]. Недостатком такого подхода является необходимость проведения длительных экспериментов на дорогостоящем оборудовании, субъективная оценка экспертом свойств исследуемой системы. Наиболее ярким примером реализации такого подхода является исследование управляемости летательного аппарата (ЛА) на динамическом стенде [24].

Численный критерий степени управляемости конкретных переменных состояния динамического объекта позволяет получить численную оценку степеней управляемости компонент вектора состояния.

Рассмотрим уравнение в канонической форме вида (21). Определим каноническую матрицу управляемости и исследуем суммы элементов каждой ее строки. Обозначим h_i суммы модулей элементов каждой строки этой матрицы, которые позволяют судить о степени управляемости компонент вектора состояния модели ЛА.

Максимальной степенью управляемости обладает компонента вектора состояния с наибольшей суммой модулей элементов соответствующей строки канонической матрицы управления h_{\max} . Степени управляемости других компонент вектора состояния определяются путем сравнения сумм модулей элементов строк канонической матрицы, соответствующих исследуемым компонентам вектора состояния с максимальным значением суммы модулей элементов канонической матрицы. Сравнить степени управляемости можно, воспользовавшись критерием следующего вида [26]:

$$\gamma = \frac{h_i}{h_{\max}}, \quad (22)$$

где h_i — модуль суммы элементов, которые находятся в i -й строке матрицы \mathbf{B}^* ; h_{\max} — максимальное значение h_i .

Предложенный критерий степени управляемости позволяет определить степень управляемости каждой конкретной компоненты вектора состояния в численном виде γ .

В соответствии с представленным численным критерием степени управляемости компонент вектора состояния мерой управляемости является конкретное число. Этот критерий удобен при автоматизированном проектировании систем управления, так как позволяет включать в вектор управления и вектор состояния только эффективно управляемые компоненты. Примером использования критерия является решение задачи выявления уча-

стков наиболее эффективного управления полетом баллистической ракеты [24].

Численный критерий степени идентифицируемости. Ставится задача оценивания неизвестных постоянных параметров матрицы объекта Φ , которая входит в уравнение объекта (10). Уравнение измерений имеет вид (11).

Вектор состояния \mathbf{x}_{n+1} можно выразить через его значение в первый момент времени следующим образом:

$$\mathbf{x}_{n+1} = \Phi^n \mathbf{x}_1 + \Phi^{n-1} \mathbf{w}_1 + \dots + \mathbf{w}_n. \quad (23)$$

Подставив выражение для \mathbf{x}_{n+1} в уравнение измерений z_{n+1} , получим

$$z_{n+1} = \mathbf{H}\Phi^n \mathbf{x}_1 + \mathbf{H}\Phi^{n-1} \mathbf{w}_1 + \dots + \mathbf{H}\mathbf{w}_n + \mathbf{v}_{n+1}. \quad (24)$$

Подставим в это уравнение выражение \mathbf{x}_1 , тогда

$$z_{n+1} = \mathbf{H}\Phi^n \begin{bmatrix} \mathbf{H} \\ \mathbf{H}\Phi \\ \mathbf{H}\Phi^2 \\ \dots \\ \mathbf{H}\Phi^{n-1} \end{bmatrix}^{-1} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ \dots \\ z_n \end{bmatrix} - \mathbf{H}\Phi^n \begin{bmatrix} \mathbf{H} \\ \mathbf{H}\Phi \\ \mathbf{H}\Phi^2 \\ \dots \\ \mathbf{H}\Phi^{n-1} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{H}\mathbf{w}_1 \\ \mathbf{H}\Phi\mathbf{w}_1 + \mathbf{H}\mathbf{w}_2 + \mathbf{v}_3 \\ \dots \\ \mathbf{H}\Phi^{n-2}\mathbf{w}_1 + \dots + \mathbf{v}_n \end{bmatrix} + \mathbf{H}\Phi^{n-1}\mathbf{w}_1 + \mathbf{H}\Phi^{n-2}\mathbf{w}_2 + \dots + \mathbf{H}\mathbf{w}_n + \mathbf{v}_{n+1}. \quad (25)$$

Введем обозначения:

$$[d_1 \ d_2 \ \dots \ d_n] = \mathbf{H}\Phi^n \begin{bmatrix} \mathbf{H} \\ \mathbf{H}\Phi \\ \mathbf{H}\Phi^2 \\ \dots \\ \mathbf{H}\Phi^{n-1} \end{bmatrix}^{-1}, \quad (26)$$

$$\begin{aligned} & \mathbf{v}_1^0 = b_1\mathbf{w}_1 + b_2\mathbf{w}_2 + \dots + \\ & + b_n\mathbf{w}_n - d_1\mathbf{v}_1 - d_2\mathbf{v}_2 - \dots - d_n\mathbf{v}_n + \mathbf{v}_{n+1} = \\ & = \mathbf{H}\Phi^n \begin{bmatrix} \mathbf{H} \\ \mathbf{H}\Phi \\ \mathbf{H}\Phi^2 \\ \dots \\ \mathbf{H}\Phi^{n-1} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{H}\mathbf{w}_1 \\ \mathbf{H}\Phi\mathbf{w}_1 + \mathbf{H}\mathbf{w}_2 + \mathbf{v}_3 \\ \dots \\ \mathbf{H}\Phi^{n-2}\mathbf{w}_1 + \dots + \mathbf{v}_n \end{bmatrix}. \quad (27) \end{aligned}$$

Таким образом, постановка задачи сводится к определению неизвестных элементов вектора-столбца $[d_1 \ d_2 \ \dots \ d_n]$ по вновь сформированным измерениям. Для $n + 2, n + 3, \dots, 2n$ моментов времени по аналогии можно записать уравнения для вновь сформированных измерений:

$$\begin{bmatrix} z_{n+1} \\ z_{n+2} \\ z_{n+3} \\ \dots \\ z_{2n} \end{bmatrix} = \begin{bmatrix} z_1 & \dots & z_n \\ z_2 & \dots & z_{n+1} \\ z_3 & \dots & z_{n+2} \\ \dots & \ddots & \dots \\ z_n & \dots & z_{2n-1} \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \dots \\ d_n \end{bmatrix} + \begin{bmatrix} v_1^0 \\ v_2^0 \\ v_3^0 \\ \dots \\ v_n^0 \end{bmatrix}, \quad (28)$$

откуда может быть выражен вектор-столбец, состоящий из неизвестных параметров d_1, d_2, \dots, d_n :

$$\begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \dots \\ d_n \end{bmatrix} = \begin{bmatrix} z_1 & \dots & z_n \\ z_2 & \dots & z_{n+1} \\ z_3 & \dots & z_{n+2} \\ \dots & \ddots & \dots \\ z_n & \dots & z_{2n-1} \end{bmatrix}^{-1} \left\{ \begin{bmatrix} z_{n+1} \\ z_{n+2} \\ z_{n+3} \\ \dots \\ z_{2n} \end{bmatrix} - \begin{bmatrix} v_1^0 \\ v_2^0 \\ v_3^0 \\ \dots \\ v_n^0 \end{bmatrix} \right\}. \quad (29)$$

Уравнение для вектора неизвестных параметров в скалярной форме будет иметь следующий вид:

$$\begin{aligned} d_1 &= f_1(z_1, \dots, z_{2n}) + v_1^{00}; \\ d_2 &= f_2(z_1, \dots, z_{2n}) + v_2^{00}; \\ &\dots \dots \dots \\ d_n &= f_n(z_1, \dots, z_{2n}) + v_n^{00}. \end{aligned} \quad (30)$$

Здесь

$$\begin{bmatrix} f_1(z_1, \dots, z_{2n}) \\ f_2(z_1, \dots, z_{2n}) \\ f_3(z_1, \dots, z_{2n}) \\ \dots \\ f_n(z_1, \dots, z_{2n}) \end{bmatrix} = \begin{bmatrix} z_1 & \dots & z_n \\ z_2 & \dots & z_{n+1} \\ z_3 & \dots & z_{n+2} \\ \dots & \ddots & \dots \\ z_n & \dots & z_{2n-1} \end{bmatrix}^{-1} \begin{bmatrix} z_{n+1} \\ z_{n+2} \\ z_{n+3} \\ \dots \\ z_{2n} \end{bmatrix},$$

$$\begin{bmatrix} v_1^{00} \\ v_2^{00} \\ v_3^{00} \\ \dots \\ v_n^{00} \end{bmatrix} = \begin{bmatrix} z_1 & \dots & z_n \\ z_2 & \dots & z_{n+1} \\ z_3 & \dots & z_{n+2} \\ \dots & \ddots & \dots \\ z_n & \dots & z_{2n-1} \end{bmatrix}^{-1} \begin{bmatrix} v_1^0 \\ v_2^0 \\ v_3^0 \\ \dots \\ v_n^0 \end{bmatrix}.$$

Точность определения параметров зависит от интенсивности приведенного измерительного шума. Приведенный измерительный шум может достигать значительных значений, что приводит к недовольному определению параметров d . Поэтому для определения неизвестных параметров целесообразно использовать алгоритмы сглаживания измерительных шумов и алгоритмы оценивания. В качестве алгоритмов оценивания можно использовать адаптивные модификации фильтра Калмана [4, 26]. В этом адаптивном алгоритме проводится оценка дисперсии измерительного шума с помощью выражения вида

$$\hat{r}_k^i = \frac{1}{2} \sqrt{\{p_0^1 k - E[(v_k^i)^2]\}^2 + 4 E[(v_k^i)^2] p_0(k-1)} - \frac{1}{2} \{p_0^i k - E[(v_k^i)^2]\}. \quad (31)$$

В практических приложениях часто необходимо знать степень идентифицируемости параметров, т.е. качественные характеристики идентифицируемых параметров: возможную точность идентификации параметров и время, за которое можно осуществить идентификацию параметра с заданной точностью.

Судить о мере идентифицируемости можно по двум характеристикам: точности идентификации и времени сходимости. Критерий, по которому определяется степень идентифицируемости, имеет вид [27]

$$\chi^i = \frac{E[(d^i)^2] r}{E[(z^i)^2] r^{*i}}, \quad (32)$$

где $E[(d^i)^2]$ — дисперсия произвольной i -й компоненты вектора параметров d^i ; $E[(z^i)^2]$ — дисперсия непосредственно измеряемой компоненты вектора параметров; $r = E[v^2]$ — дисперсия исходного измерительного шума;

$r_k^{*i} = \frac{1}{2} \sqrt{\{p_0^1 k - M[(v_k^i)^2]\}^2 + 4 M[(v_k^i)^2] p_0(k-1)} - \frac{1}{2} \{p_0^i k - M[(v_k^i)^2]\}$ — дисперсия приведенного измерительного шума.

В критерии степени идентифицируемости (32) мерой качества идентификации является скаляр. Эта особенность позволяет проводить сравнение степеней идентифицируемости компонент векторов параметров матриц различных объектов.

Таким образом, формализованная зависимость (32) используется для определения степени идентифицируемости параметров матрицы Φ . Дисперсия исходного измерительного шума определяется из практических соображений в соответствии с режимом работы измерительной системы или принимается значение из паспорта измерительного

прибора. Определенные сложности возникают при вычислении приведенного измерительного шума. Однако при использовании при идентификации параметров матрицы модели исследуемой системы скалярного адаптивного алгоритма оценивания дисперсия приведенного измерительного шума вычисляется на каждом шаге функционирования алгоритма.

Качество идентификации или эффективность идентификации определяется максимально достижимой точностью идентификации и необходимым временем достижения заданной точности идентификации.

Заключение

Исследованные численные критерии степени наблюдаемости, управляемости и идентифицируемости имеют ясный физический смысл, отличаются простотой, универсальностью, позволяют определять качество наблюдения и управления компонент вектора состояния, а также идентификации параметров в виде скаляра, что чрезвычайно удобно при использовании в практических приложениях.

Можно констатировать, что свойство дуальности в исследованных критериях степени наблюдаемости и управляемости явно не проявляется. Интуитивно ясно, что имеется связь между степенью управляемости и запасом устойчивости, степенью наблюдаемости и степенью идентифицируемости, но для получения формализованных зависимостей в явном виде требуются дополнительные исследования. Систематизация результатов подобных исследований существенно дополнит теорию качественного анализа систем.

Список литературы

1. **Воронов А. А.** Устойчивость, управляемость, наблюдаемость. М.: Наука, 1979.
2. **Эйкхофф П.** Основы идентификации систем управления. М.: Мир, 1975.
3. **Брайсон, Хо Ю Ши.** Прикладная теория оптимального управления. М.: Мир, 1972.
4. **Кузовков Н. Т., Карабанов С. В., Салычев О. С.** Непрерывные и дискретные системы управления и методы идентификации. М.: Машиностроение, 1978.
5. **Балонин Н. А.** Теоремы идентифицируемости. СПб.: Политехника, 2010.
6. **Балакришнан А. В.** Теория фильтрации Калмана. М.: Мир, 1988.
7. **Айвазян С. А., Енюков И. С., Мешалкин Л. Д.** Прикладная статистика: Исследование зависимостей. М.: Финансы и статистика, 1985.
8. **Yusupov R., Rozenwasser E.** Sensitivity of Automatic Control Systems. London: CRS Press, 1999.

9. **Ивановский Р. И., Игнатов А. А.** Теория чувствительности в задачах управления и оценки. СПб.: ЦНИИ "РУМБ", 1986.
10. **Шмидт С. В., Белова Д. Ю., Калиев Б. З.** Применение функции чувствительности к энергетическим задачам // Он-лайн Электрик: Электроэнергетика. Новые технологии, 2012. URL: <http://www.jgline-electric.ru/articles.php?id=30> (дата обращения: 10.05.15).
11. **Ткаченко В. Н., Коротков Е. К., Поздняков Е. К.** Исследование исходных параметров метода определения дальности целей в пассивных многопозиционных системах при помощи функции чувствительности // Изв. ЮФУ. Технические науки, 2014. № 8 (157). С. 170–177.
12. **Парусников Н. А., Морозов В. М., Борзов В. И.** Задача коррекции в инерциальной навигации. М.: МГУ, 1982.
13. **Brown R. G.** Not just observable, but how observable? // National Electronics Conference Proceedings. 1966. N. 22. P. 409–714.
14. **Ablin H. L.** Criteria for degree of observability in a control system // Retrospective Theses and Dissertations. Paper 3188. Iowa State University, 1967.
15. **Ham F. M., Brown R. G.** Observability, eigenvalues, and Kalman filtering // IEEE Trans. on Aerospace and Electronic Systems. 1983. Vol. AES-19, Iss. 2. P. 269–273.
16. **Kai Shen, Neusyupin K. A., Proletarsky A. V.** On State Estimation of Dynamic Systems by Applying Scalar Estimation Algorithms // Proc. of 2014 IEEE Chinese Guidance, Navigation and Control Conference. 2014. August 8–10. Yantai, China. P. 124–129.
17. **Кай Шэнь, Пролетарский А. В., Неусыпин К. А.** Исследование степени наблюдаемости погрешностей автономных инерциальных навигационных систем // Автоматизация и современные технологии. 2015. № 1. С. 24–30.
18. **Зубов Н. Е., Микрин Е. А., Мисриханов М. Ш.** и др. Идентификация положения равновесной ориентации международной космической станции как задача матричного пополнения с устойчивостью // Изв. РАН. Теория и системы управления. 2012. № 2. С. 130–144.
19. **Мисриханов М. Ш., Рябченко В. Н.** Алгебраические и матричные методы в теории линейных МИМО-систем // Вестник ИГЭУ. 2005. Вып. 5. С. 196–240.
20. **Салычев О. С.** Скалярное оценивание многомерных динамических систем. М.: Машиностроение, 1987.
21. **Афанасьев В. Н., Неусыпин К. А.** Навигационный комплекс. Патент на изобретение RUS 2016383.
22. **Афанасьев В. Н., Неусыпин К. А.** Синтез адаптивного регулятора инерциальной навигационной системы // Изв. РАН. Техническая кибернетика. 1992. № 2. С. 178–183.
23. **Kalman R. E., Ho Y. C., Narendra K. S.** Controllability of linear dynamical systems. // Contributions to the Theory of Differential Equations. 1963. Vol. I, No. 2. P. 189–213.
24. **Богословский С. В., Дорофеев А. Д.** Динамика полета летательных аппаратов. СПб. ГУАП, СПб., 2002.
25. **Кай Шэнь.** Разработка методов оценивания и прогноза навигационных систем летательных аппаратов // Автоматизация. современные технологии. 2015. № 7. С. 13–18.
26. **Неусыпин К. А., Фам Суан Фанг.** Численный критерий степени управляемости переменных состояния // Автоматизация и современные технологии. 2007. № 7. С. 24–26.
27. **Неусыпин К. А., Пролетарский А. В., Кузнецов И. А.** Синтез численного критерия меры идентифицируемости параметров моделей // Автоматизация. Современные технологии. 2015. № 3. С. 9–13.

Study of the Criteria for the Degrees of Observability, Controllability and Identifiability of the Linear Dynamical Systems

Kai Shen, shenkaichn@mail.ru, Nanjing University of Science and Technology
K. A. Neusypin, neusypin@mail.ru, Bauman Moscow State Technical University

Corresponding author: K. A. Neusypin, D. Sc., Professor,
Bauman Moscow State University, Moscow, 105005, Russian Federation
e-mail: neusypin@mail.ru

Received on June 20, 2016

Accepted on July 08, 2016

The article is devoted to approaches to determination of the degree of observability, controllability and identifiability of the parameters in the linear dynamical systems. Well-known criteria for determination of the observability, controllability and identifiability of the dynamical objects were presented. Those criteria only allow us to determine, which component of the state vector is observable or controllable better. They can only give a relative assessment of the quality of the observability and controllability of a specific component of the state vector of the system, whereas, it is impossible to compare the quantitative characteristics of the state variables and parameters in various systems. Numerical criteria for calculation of the degree of observability and controllability of the state variables, as well as the degree of identifiability of the model parameters of the dynamical objects were researched using the scalar methods. The proposed numerical criteria have a clear physical meaning, and are characterized by simplicity and versatility. They allow us to determine the quality of the observability, controllability and identifiability in a scalar form, which can be very convenient for practical applications.

Keywords: model of dynamical object, degree of identifiability, degree of controllability, degree of observability

Acknowledgements: This work was supported by the Programme of Introducing Talents of Discipline to Universities in P. R. China ("111 program" No. B 16025).

For citation:

Kai Shen, Neusypin K. A. Study of the Criteria for the Degrees of Observability, Controllability and Identifiability of the Linear Dynamical Systems, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2016, vol. 17, no. 9, pp. 723–731.

DOI: 10.17587/mau.17.723-731

References

1. Voronov A. A. *Ustojchivost', upravlyaemost', nablyudaemost'* (Stability, controllability, observability), Moscow, Nauka, 1979 (in Russian).
2. E'jks'off P. *Osnovy identifikacii sistem upravleniya* (Fundamentals of identification of control systems), Moscow, Mir, 1975 (in Russian).
3. Brajson, Xo Yu Shi. *Prikladnaya teoriya optimal'nogo upravleniya* (Applied optimal control theory), Moscow, Mir, 1972 (in Russian).
4. Kuzovkov N. T., Karabanov S. V., Salychev O. S. *Nep'ryvnye i diskretnye sistemy upravleniya i metody identifikacii* (Continuous and discrete control systems and identification methods), Moscow, Mashinostroenie, 1978 (in Russian).
5. Balonin N. A. *Teoremy identifikacii* (Theorem of identifiability), Saint Petersburg, Politehnika, 2010 (in Russian).
6. Balakrishnan A. V. *Teoriya fil'tracii Kalmana* (Theory of Kalman filter), Moscow, Mir, 1988 (in Russian).
7. Ajvazyan S. A., Enyukov I. S., Meshalkin L. D. *Prikladnaya statistika: Issledovanie zavisimostej* (Applied statistics: Research of dependences), Moscow, Finansy i statistika, 1985 (in Russian).
8. Yusupov R., Rozenwasser E. *Sensitivity of Automatic Control Systems*, London, CRS Press, 1999.
9. Ivanovskiy R. I., Ignatov A. A. *Teoriya chuvstvitel'nosti v zadachax upravleniya i ocenki* (Theory of sensitivity in the task control and estimation), Saint Petersburg, TsNII "RUMB", 1986 (in Russian).
10. Schmid C. V., Belova D. Yu., Kaliev B. Z. *Primenenie funkcii chuvstvitel'nosti k e'nergeticheskim zadacham* (Application of the sensitivity function to the energy task), Onlajn E'lektrik: E'lektroe'nergetika, Noveye tekhnologii, 2012, available at: <http://www.jgline-electric.ru/articles.php?id=30> (date of access: 10.05.15) (in Russian).
11. Tkachenko V. N., Korotkov E. K., Pozdnyakov E. K. *Issledovanie isxodnyx parametrov metoda opredeleniya dal'nosti celej v passivnyx mnogopozicionnyx sistemax pri pomoshhi funkcii chuvstvitel'nosti* (Research of the input parameters from the target distance determination method in the passive multiposition system by using the sensitivity function), *Izv. YuFU, Texnicheskie nauki*, 2014, no. 8, pp. 170–177 (in Russian).
12. Parusnikov N. A., Morozov V. M., Borzov V. I. *Zadacha korr'ekcii v inerci-al'noj navigacii* (The correction problem of inertial navigation), Moscow, MGU, 1982 (in Russian).
13. Brown R. G. Not just observable, but how observable? *National Electronics Conference Proceedings*, 1966, no. 22, pp. 409–714.
14. Ablin H. L. Criteria for degree of observability in a control system, *Retrospective Theses and Dissertations*, Paper 3188, Iowa State University, 1967.
15. Ham F. M., Brown R. G. Observability, eigenvalues, and Kalman filtering, *IEEE Trans. on Aerospace and Electronic Systems*, 1983, vol. AES-19, iss. 2, pp. 269–273.
16. Kai Shen, Neusypin K. A., Proletarskiy A. V. On State Estimation of Dynamic Systems by Applying Scalar Estimation Algorithms, *Proc. of 2014 IEEE Chinese Guidance, Navigation and Control Conference*, August 8–10, 2014, Yantai, China, pp. 124–129.
17. Kai Shen, Proletarskiy A. V., Neusypin K. A. *Issledovanie stepeni nablyudaemosti pogreshnostej avtonomnykh inercial'nykh navigacionnykh sistem* (Research of the degree of observability of errors of autonomous inertial navigation systems), *Avtomatizatsiya i Sovremennye Tekhnologii*, 2015, no. 1, pp. 24–30 (in Russian).
18. Zubov N. E., Mikrin E. A., Misrixanov M. Sh. i dr. *Identifikaciya polozheniya ravnovesnoj orientacii mezhdunarodnoj kosmicheskoy stancii kak zadacha matrichnogo popolneniya s ustojchivost'yu* (Identification of the equilibrium orientation of the international space station as a problem of matrix replenishment stability), *Izv. RAN. Teoriya i Sistemy Upravleniya*, 2012, no. 2, pp. 130–144 (in Russian).
19. Misrixanov M. Sh., Ryabchenko V. N. *Algebraicheskie i matrichnye metody v teorii linejnykh MIMO-sistem* (The algebraic and matrix methods in the theory of linear MIMO-systems), *Vestnik IGE'U*, 2005, vol. 5, pp. 196–240 (in Russian).
20. Salychev O. S. *Skalyarnoe ocenivanie mnogomernykh dinamicheskix sistem* (Scalar estimation of multidimensional dynamical systems), Moscow, Mashinostroenie, 1987 (in Russian).
21. Afanas'ev V. N., Neusypin K. A. *Navigacionnyj kompleks* (Navigation systems), Patent na izobrenenie RUS 2016383 (in Russian).
22. Afanas'ev V. N., Neusypin K. A. *Sintez adaptivnogo regulyatora inercial'noj navigacionnoj sistemy* (The synthesis of adaptive controller for inertial navigation systems), *Izv. RAN. Texnicheskaya Kibernetika*, 1992, no. 2, pp. 178–183 (in Russian).
23. Kalman R. E., Ho Y. C., Narendra K. S. Controllability of linear dynamical systems, *Contributions to the Theory of Differential Equations*, 1963, vol. 1, no. 2, pp. 189–213.
24. Bogoslovskiy S. V., Dorofeev A. D. *Dinamika poleta letatel'nykh apparatov* (Flight dynamics of aircrafts), Saint Petersburg, GUAP, SPb., 2002 (in Russian).
25. Kai Shen. *Razrabotka metodov ocenivaniya i prognoza navigacionnykh sistem letatel'nykh apparatov* (Development of estimation and prediction methods of aircraft navigation systems), *Avtomatizatsiya. Sovremennye Tekhnologii*, 2015, no. 7, pp. 13–18 (in Russian).
26. Neusypin K. A., Fam Suan Fang. *Chislennyj kriterij stepeni upravlyaemosti peremennykh sostoyaniya* (The numerical criterion of the degree of controllability of state variables), *Avtomatizatsiya i Sovremennye Tekhnologii*, 2007, no. 7, pp. 24–26 (in Russian).
27. Neusypin K. A., Proletarskiy A. V., Kuznecov I. A. *Sintez chislennogo kriteriya mery identifikacii parametrov modelej* (The synthesis of numerical criterion for the degree of identifiability of model parameters), *Avtomatizatsiya. Sovremennye Tekhnologii*, 2015, no. 3, pp. 9–13 (in Russian).

Е. М. Васильев, канд. техн. наук, доц., **А. С. Одношивкин**, магистрант, vgtu-aits@yandex.ru,
Воронежский государственный технический университет

Гомеостатическое управление динамическими системами на основе нечетких регуляторов

Рассматривается задача гомеостатического управления системами с существенной неопределенностью динамической структуры и нестационарностью параметров. На примере процесса двухконтурного противоточного теплообмена раскрываются приемы и способы решения поставленной задачи на основе нечетких регуляторов. Приводятся результаты численного моделирования синтезированной системы управления и обсуждаются ее свойства.

Ключевые слова: гомеостатическое управление, многоуровневые и многорежимные системы, принципы многоальтернативности, нечеткие алгоритмы

Введение

К характерным особенностям задач управления сложными динамическими системами относятся [1]:

- существенная неопределенность и высокий порядок дифференциальных уравнений, описывающих эти системы;
- нестационарность во времени параметров системы и внешних условий;
- распределенность процессов в пространстве, приводящая к значительному транспортному запаздыванию управляющих воздействий.

Вытекающая из этих особенностей необходимость упрощения моделей объектов и методик расчета соответствующих систем управления в общем случае затрудняет обеспечение показателей качества этих систем и приводит к росту опасности неуправляемого протекания процессов в критических условиях. В связи с этим актуальной остается проблема поиска подходов и методов управления, совмещающих в себе простоту реализации и надежность достижения целей управления в широком диапазоне режимов функционирования систем, в том числе и в критических режимах.

В предлагаемой статье рассматривается возможность решения указанной проблемы на основе концепции гомеостатического управления, сформулированной У. Эшби [2] и сохраняющей свою актуальность в настоящее время [3–7].

В основе этой концепции лежит базовая кибернетическая предпосылка об искусственном воспроизведении способов функционирования биологических систем, существующих в условиях изменяющейся внешней среды и влияния естественного отбора. Успешное развитие живых организмов в указанных неопределенных условиях, требующих от популяций непрерывной борьбы за свое существование, свидетельствует о том, что в процессе эволюции в биологических сообществах сформировались некоторые общие свойства, обеспечивающие им возможность этого развития, в частности, *свойство гомеостаза*.

Биологическое содержание гомеостаза состоит в способности организма поддерживать характеристики своего функционирования в физиологически допустимых пределах [3, 4, 6]. Техническая реализация этого свойства в искусственных системах основана на ряде достаточно универсальных принципов, в качестве которых постулируются [2, 6–10]:

- принцип многоуровневости (иерархичности) управления, реализуемый путем организации в открытых системах, по меньшей мере, двух уровней управления, один из которых функционирует в области допустимых вариаций состояния системы, а другие реагируют только на критические отклонения от этого состояния;
- принцип многообразия алгоритмов управления и разделения функций (многорежимность и переключение структур), тесно связанный с принципом многоуровневости и отражающий гибкость алгоритмов управления в изменяющихся условиях функционирования системы [11–12];
- принцип модульности (дискретности) строения, обеспечивающий разнообразие уровней и режимов управления на основе комбинаций ограниченного набора элементарных модулей системы [13–14].

Совокупность сформулированных принципов позволяет решать поставленные выше задачи управления не в результате качественного усложнения соответствующих алгоритмов, но путем количественного наращивания в системе числа простых, альтернативных способов управления, отвечающих информационному принципу необходимого разнообразия: "Только разнообразие может уничтожить разнообразие" [15].

Рассмотрим реализацию этих принципов на примере системы управления процессом теплообмена, являющегося типовым образцом многомерной открытой системы [16]. В работе использована реализация алгоритмов управления на основе нечетких регуляторов, выбор которых рассматривается ниже.

Описание объекта управления

В качестве объекта управления выбран двухконтурный процесс противоточного теплообмена, обобщенная схема которого показана на рис. 1.

На схеме обозначены: $T_{x, \text{ВХ}}, T_{x, \text{ВЫХ}}$ — температуры "холодного" теплоносителя вторичного контура на входе и выходе теплообменника соответственно; $T_{\Gamma, \text{ВХ}}, T_{\Gamma, \text{ВЫХ}}$ — температуры "горячего" теплоносителя контура нагревателя на входе и выходе теплообменника; $T_{\text{Н, ВХ}}, T_{\text{Н, ВЫХ}}$ — температуры теплоносителя на входе и выходе нагревателя; θ — время, необходимое для прохождения теплоносителем расстояния между теплообменником и нагревателем; $e^{-p\theta}$ — передаточная функция звена запаздывания; g_x, g_Γ — массовые расходы теплоносителей в секунду в контуре потребителей и в контуре нагревателя; U — температура рабочей поверхности нагревателя.

Объект управления дополнительно введен нагреватель, образующий совместно с теплообменником замкнутый первичный контур общего процесса теплообмена (рис. 1).

Нагреватель и теплообменник представляются отдельными цепочками из N последовательных ячеек, внутри каждой из которых температура теплоносителей в течение заданного малого интервала dt времени принимается постоянной. Такая пространственная и временная дискретизация рассматриваемых процессов позволяет описать их в разностной форме [17].

Для ячейки j ($j = \overline{1, N}$) нагревателя на интервале времени i :

$$\frac{dT_{\text{Н}, i-1}^j}{dt} = \frac{[ks(U - T_{\text{Н}, i-1}^j) - g_\Gamma c(T_{\text{Н}, i-1}^j - T_{\text{Н}, i-1}^{j-1})]}{m_\Gamma c}, \quad (1)$$

$$T_{\text{Н}, i}^j = T_{\text{Н}, i-1}^j + \left(\frac{dT_{\text{Н}, i-1}^j}{dt} \right) dt, \quad (2)$$

для ячейки j теплообменника (значение j возрастает по направлению движения "холодного" теплоносителя):

$$\begin{aligned} \frac{dT_{x, i-1}^j}{dt} &= \\ &= \frac{[ks(T_{\Gamma, i-1}^j - T_{x, i-1}^j) - g_x c(T_{x, i-1}^j - T_{x, i-1}^{j-1})]}{m_x c}, \quad (3) \end{aligned}$$

$$\begin{aligned} \frac{dT_{\Gamma, i-1}^j}{dt} &= \\ &= \frac{[-ks(T_{\Gamma, i-1}^j - T_{x, i-1}^j) + g_\Gamma c(T_{\Gamma, i-1}^{j+1} - T_{\Gamma, i-1}^j)]}{m_\Gamma c}, \quad (4) \end{aligned}$$

$$T_{x, i}^j = T_{x, i-1}^j + \left(\frac{dT_{x, i-1}^j}{dt} \right) dt, \quad (5)$$

$$T_{\Gamma, i}^j = T_{\Gamma, i-1}^j + \left(\frac{dT_{\Gamma, i-1}^j}{dt} \right) dt, \quad (6)$$

$$T_{\Gamma, \text{ВХ}, i} = T_{\text{Н, ВЫХ}, i - \frac{\theta}{dt}}; \quad T_{\text{Н, ВХ}, i} = T_{\Gamma, \text{ВЫХ}, i - \frac{\theta}{dt}}, \quad (7)$$

где $\frac{dT}{dt}$ — градиент температуры в элементарной ячейке; c — удельная теплоемкость теплоносителя; k — коэффициент теплопередачи через стенку между горячим и холодным теплоносителями; s — площадь теплопередачи в пределах одной ячейки; m_Γ, m_x — массы теплоносителей в элементарной ячейке; $T_{\text{Н}}^0 = T_{\text{Н, ВХ}}, T_{\text{Н}}^N = T_{\text{Н, ВЫХ}}, T_{\text{Х}}^0 = T_{\text{Х, ВХ}}, T_{\text{Х}}^N = T_{\text{Х, ВЫХ}}, T_{\Gamma}^{N+1} = T_{\Gamma, \text{ВХ}}, T_{\Gamma}^1 = T_{\Gamma, \text{ВЫХ}}$.

Численное решение уравнений (1)–(6) реализовано в пакете MatLab на имитационной модели, общий вид которой представлен на рис. 2.

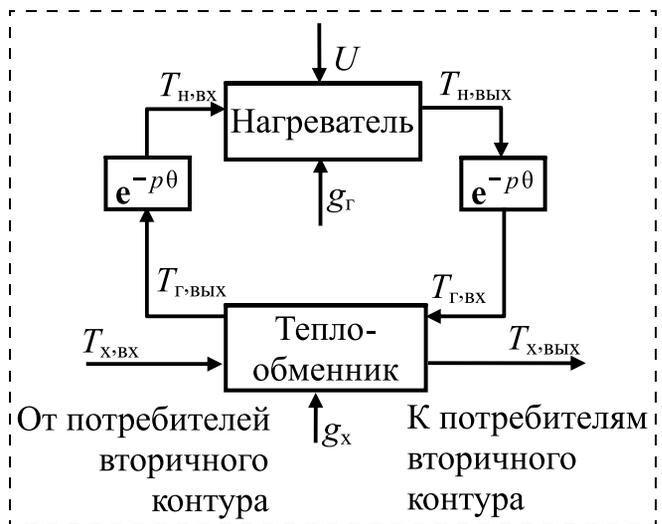


Рис. 1. Схема противоточного теплообмена

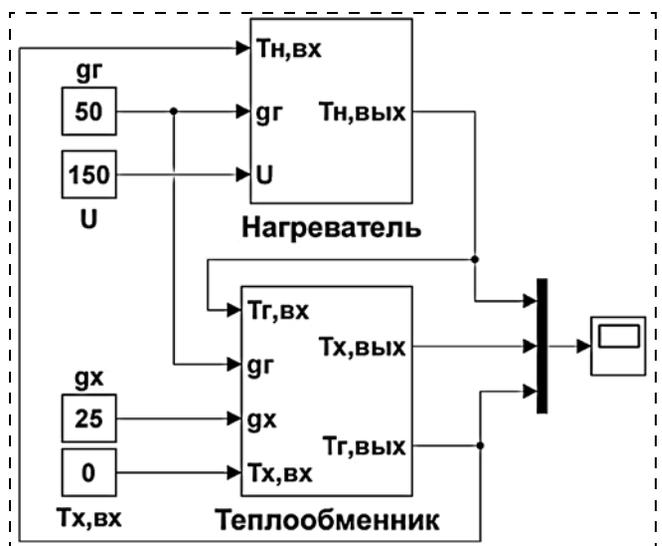


Рис. 2. Имитационная модель процесса теплообмена

Отличительной особенностью этой модели является прямая имитация функционирования каждой из N идентичных ячеек нагревателя и теплообменника с шагом решения dt , принятым при временной дискретизации процесса. Фрагмент модели нагревателя с двумя первыми ячейками ($j = 1, 2$) показан на рис. 3.

Программируемые модули "cell" на рис. 3 содержат операторы, соответствующие уравнениям (1) и (2). В модели теплообменника аналогичные операторы реализуют уравнения (3)–(7).

На рис. 4 показаны переходные процессы в теплообменнике, построенные с помощью модели (1)–(7).

Числовые значения параметров моделирования: $m_r = 25$ кг; $m_x = 30$ кг; $c = 4190$ Дж/кг·К; $k = 2500$ Вт/м²·К; $g_x = 25$ кг/с; $g_r = 50$ кг/с; $N = 100$; $s = 0,3$ м²; $\theta = 0$ с; $dt = 0,1$ с.

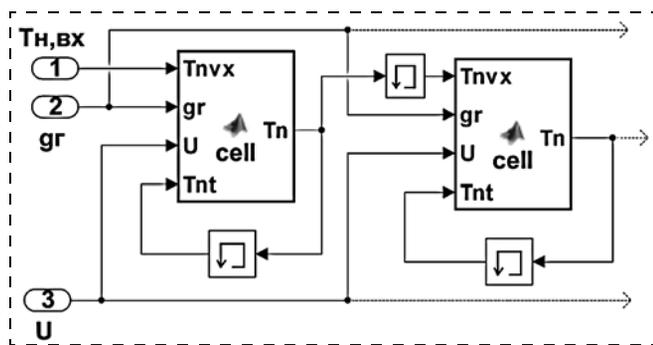


Рис. 3. Фрагмент модели нагревателя

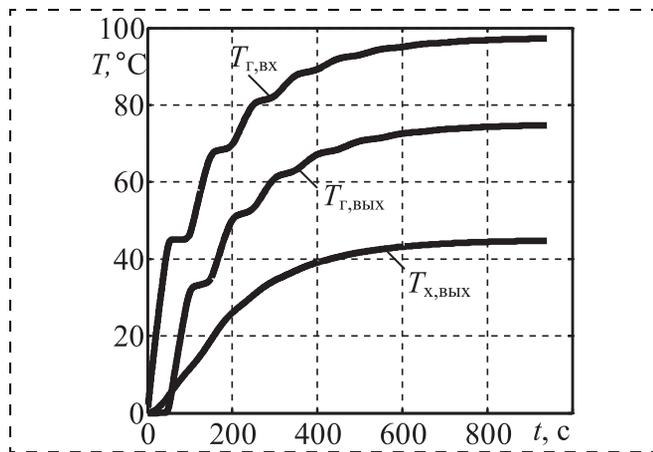


Рис. 4. Переходные процессы в теплообменнике

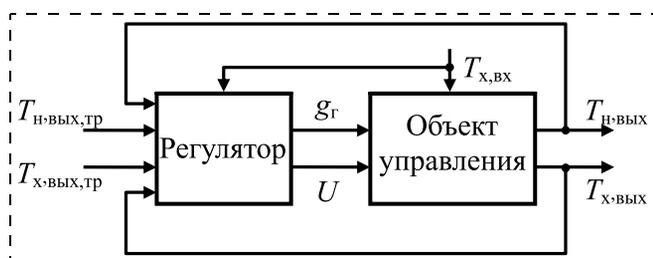


Рис. 5. Функциональная схема системы управления

Начальные температуры теплоносителей в системе приняты нулевыми. Температура нагревателя $U = 150$ °С поддерживалась постоянной. На рис. 4 хорошо видна 100-секундная периодичность $T_{Г, ВХ}$ и $T_{Г, ВЫХ}$, характерная для замкнутого цикла в пространственно-распределенном контуре нагревателя.

Система гомеостатического управления

В качестве координат состояния системы будем использовать:

- регулируемые величины: $T_{Х, ВЫХ}$, $T_{Н, ВЫХ}$ — температуры теплоносителей на выходе теплообменника и на выходе нагревателя (см. рис. 1), значение $T_{Х, ВЫХ}$ должно поддерживаться в диапазоне $T_{Х, ВЫХ} = 60...70$ °С, а $T_{Н, ВЫХ}$ — не превышать некоторого максимального значения $T_{Н, ВЫХ, МАКС}$, например, $T_{Н, ВЫХ, МАКС} = 100$ °С;
- управляющие воздействия: U , g_r — температура рабочей поверхности нагревателя и массовый расход теплоносителя в секунду в контуре нагревателя;
- возмущающее воздействие: $T_{Х, ВХ}$ — температура теплоносителя на входе теплообменника, изменяющаяся в диапазоне $30...50$ °С со скоростью до $0,8$ °С/мин.

Функциональная схема системы управления показана на рис. 5, где $T_{Н, ВЫХ, ТР}$ — требуемое среднее значение температуры теплоносителя на выходе нагревателя; $T_{Х, ВЫХ, ТР}$ — требуемое среднее значение температуры теплоносителя вторичного контура на выходе теплообменника.

Исходя из априорной неопределенности и нестационарности параметров объекта, будем синтезировать регуляторы в классе нечеткого управления. Дополнительным аргументом в пользу этого выбора послужила отмеченная в работе [18] исключительная возможность нечеткого управления: формировать независимые индивидуальные правила регулирования для различных состояний объекта управления. Такая возможность хорошо соответствует заявленным выше принципам гомеостатического управления.

В соответствии с принципом многоуровневости сформируем структуру системы автоматического управления с пятью уровнями регулирования в каждом канале воздействий g_r и U .

Структурная схема канала управления по g_r показана на рис. 6.

На основе текущего значения контролируемой температуры $T_{Н, ВЫХ}$ формируется лингвистическая переменная $\tilde{T}_{Н, ВЫХ} = \gamma$, где γ содержит нечеткие переменные:

$$\gamma = \{\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5\},$$

например, γ_1 — температура $T_{Н, ВЫХ}$ существенно ниже требуемого среднего значения $T_{Н, ВЫХ, ТР}$; γ_2 — температура $T_{Н, ВЫХ}$ ниже требуемого среднего значения $T_{Н, ВЫХ, ТР}$; γ_3 — $T_{Н, ВЫХ}$ примерно равна $T_{Н, ВЫХ, ТР}$; γ_4 —

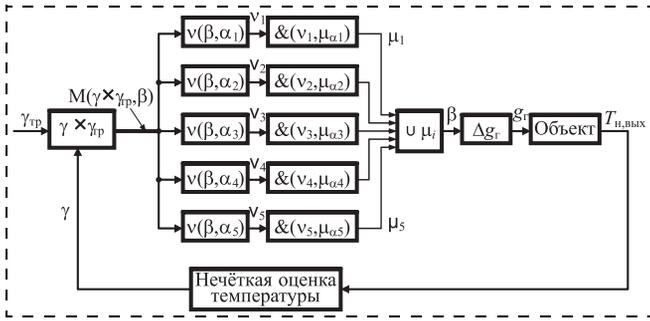


Рис. 6. Структурная схема канала управления по g_r

$T_{н, вых}$ выше требуемой; γ_5 — $T_{н, вых}$ существенно выше требуемой температуры $T_{н, вых, тр}$.

На рис. 7 приведены используемые в примере функции принадлежности $\mu_{\gamma 1}(T_{н, вых}), \dots, \mu_{\gamma 5}(T_{н, вых})$.

Приращению Δg_r массового расхода g_r поставим в соответствие лингвистическую переменную α :

$$\alpha = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5\},$$

со значениями, используемыми, например, меры сравнения: α_1 — расход g_r теплоносителя следует существенно уменьшить по сравнению с его текущим значением; ... α_3 — расход g_r практически не изменять; ...; α_5 — расход g_r следует существенно увеличить. Используемые функции принадлежности $\mu_{\alpha 1}(\Delta g_r), \dots, \mu_{\alpha 5}(\Delta g_r)$ показаны на рис. 8.

Для каждой нечеткой переменной $\alpha_1, \dots, \alpha_5$ в структуре системы выделен отдельный канал управления, вклад каждого из них в результирующее нечеткое управляющее воздействие β , вырабатываемое регулятором, определяется мерой близости текущего состояния γ к требуемому состоянию $\gamma_{тр}$:

$$\gamma_{тр} = \{\mu_{\gamma 1}(T_{н, вых, тр}), \mu_{\gamma 2}(T_{н, вых, тр}), \dots, \mu_{\gamma 5}(T_{н, вых, тр})\}. \quad (8)$$

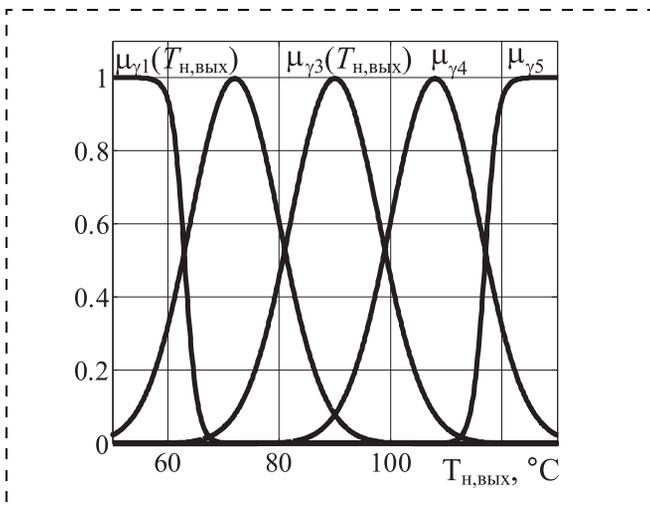


Рис. 7. Функции принадлежности нечетких переменных $\gamma_1, \dots, \gamma_5$

В общем случае $T_{н, вых, тр} = f(T_{х, вх})$, и с ростом $T_{х, вх}$ значение $T_{н, вых, тр}$ снижается.

Используя декартово произведение множеств γ и $\gamma_{тр}$, получим матрицу перехода $M(\gamma \times \gamma_{тр}, \beta)$ из текущего состояния γ объекта в требуемое $\gamma_{тр}$ под действием искомого управления β :

$$M(\gamma \times \gamma_{тр}, \beta) = \gamma \times \gamma_{тр}. \quad (9)$$

Для определения общего управления β для каждого из n каналов системы экспертным путем формируются матрицы перехода $M(\gamma_i \times \gamma_j, \alpha_r)$, $i = 1, \dots, n$; $j = 1, \dots, n$; $r = 1, \dots, n$; $n = 5$, значения элементов которых характеризуют возможность перехода объекта из состояния γ_i в состояние γ_j под действием нечеткого управления α_r .

Примеры таких матриц для управлений α_1 и α_2 имеют вид:

$$M(\gamma_i \times \gamma_j, \alpha_1) =$$

γ_i	γ_j				
	γ_1	γ_2	γ_3	γ_4	γ_5
γ_1	0,50	0,80	1	0,80	0,5
γ_2	0,15	0,50	0,80	1	0,8
γ_3	0,05	0,15	0,50	0,80	1
γ_4	0,01	0,05	0,15	0,50	1
γ_5	0,005	0,01	0,05	0,15	1

$$M(\gamma_i \times \gamma_j, \alpha_2) =$$

γ_i	γ_j				
	γ_1	γ_2	γ_3	γ_4	γ_5
γ_1	0,80	1	0,80	0,50	0,15
γ_2	0,50	0,80	1	0,80	0,50
γ_3	0,15	0,50	0,80	1	0,80
γ_4	0,05	0,15	0,50	0,80	1
γ_5	0,01	0,05	0,15	0,50	1

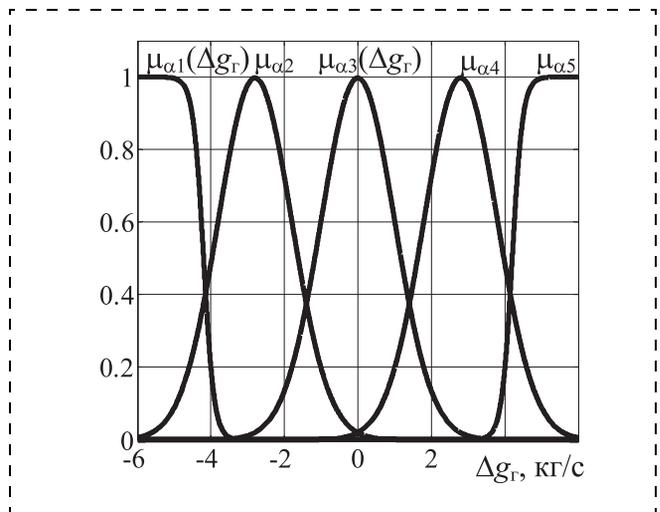


Рис. 8. Функции принадлежности нечетких переменных $\alpha_1, \dots, \alpha_5$

В качестве меры близости искомого управления β к известным управлениям $\alpha_1, \dots, \alpha_5$ применяется степень включения матрицы перехода $M(\gamma \times \gamma_{TP}, \beta)$ в соответствующие матрицы для каждого канала:

$$v_r = v(\beta, \alpha_r) = v[M(\gamma \times \gamma_{TP}, \beta), M(\gamma_i \times \gamma_j, \alpha_r)] = \min\{\max[1 - M(\gamma \times \gamma_{TP}, \beta)_{ij}, M(\gamma_i \times \gamma_j, \alpha_r)_{ij}]\}. \quad (10)$$

Полученные значения мер близости v_r используются для логического вывода:

$$\mu_r(\Delta g_r) = \&_{\Delta g_r}(v_r, \mu_{\alpha_r}(\Delta g_r)), \quad (11)$$

результатом $\mu_r(\Delta g_r)$ которого являются усеченные на уровне v_r функции принадлежности $\mu_{\alpha_r}(\Delta g_r)$.

Управление β может быть определено объединением по всем нечетким подмножествам $\mu_r(\Delta g_r)$:

$$\beta = \{\mu(\Delta g_r)\} = \left\{ \bigcup_{\Delta g_r} \mu_r(\Delta g_r) \right\}. \quad (12)$$

Вычисление текущего четкого значения $\Delta g_r(t)$ осуществляется методом центраида:

$$\Delta g_r(t) = \frac{\int \Delta g_r \mu(\Delta g_r) d\Delta g_r}{\int \mu(\Delta g_r) d\Delta g_r}, \quad (13)$$

с последующим интегрированием по времени для получения управляющего воздействия $g_r(t)$:

$$g_r(t) = \int \Delta g_r(t) dt. \quad (14)$$

Процедуры (8)–(14) определяют собой алгоритм работы системы в канале управления температурой $T_{н, вых}$ теплоносителя на выходе нагревателя с помощью изменения массового расхода g_r этого теплоносителя в первичном контуре. Отметим, что для

формирования управления (8)–(14) не потребовалась информация о динамической модели объекта.

Структура регулятора канала управления выходной температурой $T_{х, вых}$ теплоносителя вторичного контура с помощью изменения температуры U поверхности нагревателя аналогична рассмотренной выше и также содержит пять уровней управления.

Результаты численного моделирования

Численный эксперимент, проведенный на модели системы (1)–(14), предусматривал проверку работоспособности трех последовательно усложняющихся вариантов ее структуры, позволяющих получить наглядное сопоставление результативности каждого принимаемого решения.

На рис. 9 показаны переходные процессы в структуре системы, содержащей только контур регулирования температуры $T_{х, вых}$ вторичного теплоносителя на выходе теплообменника (см. также рис. 5).

Изменения $T_{х, вых}$ на рис. 9 удерживаются системой в заданных пределах 60...70 °С, однако ошибка регулирования при изменяющемся во времени возмущении не может быть принципиально устранена средствами рассматриваемого контура. Для подавления системой скоростной заводки (скоростного всплеска) возмущения $T_{х, вх}$ в структуре организован альтернативный канал ввода $T_{х, вх}$, вырабатывающий сигнал $w \dot{T}_{х, вх}$, пропорциональный скорости $\dot{T}_{х, вх}$ и вычитаемый из $T_{х, вых}$, тр на входе регулятора (см. рис. 5). Очевидно, что этот

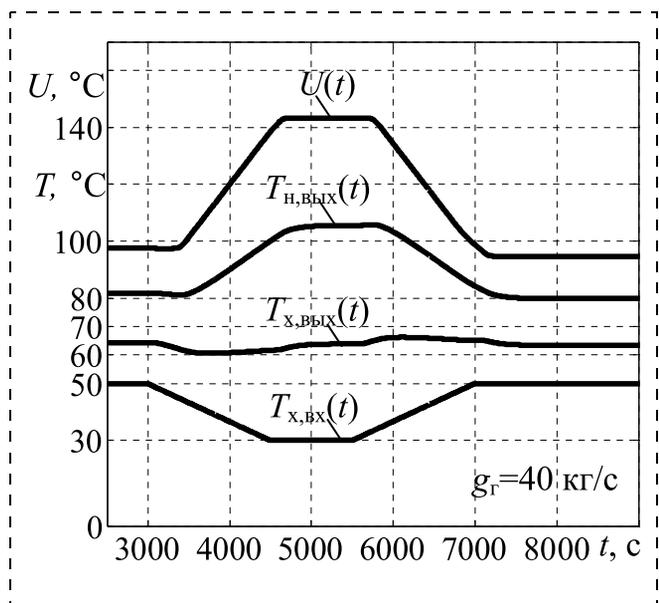


Рис. 9. Переходные процессы в системе с контуром регулирования $T_{х, вых}$

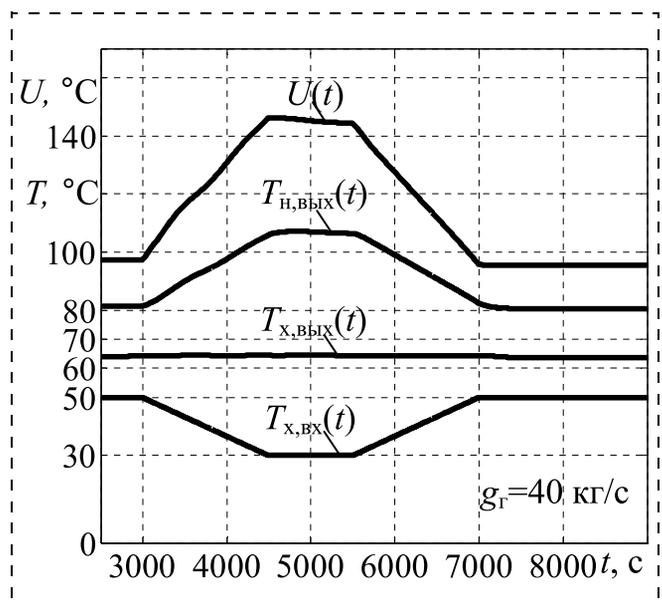


Рис. 10. Переходные процессы в системе с альтернативным каналом ввода возмущения

канал функционирует только в режимах с изменяющейся $T_{x, \text{вх}}$, т.е. когда $\dot{T}_{x, \text{вх}} \neq 0$.

Результат работы альтернативного канала показан на рис. 10 и подтверждает обоснованность его введения.

На рис. 10 (на интервале $t = 3000 \dots 4500$ с) можно заметить последовательную передачу управления между каналами регулятора по изменению наклона управляющего воздействия U в процессе отработки изменяющегося возмущения $T_{x, \text{вх}}$.

Третий вариант структуры предусматривает введение контура регулирования температуры $T_{n, \text{вых}}$ первичного теплоносителя на выходе нагревателя (см. рис. 5 и рис. 11).

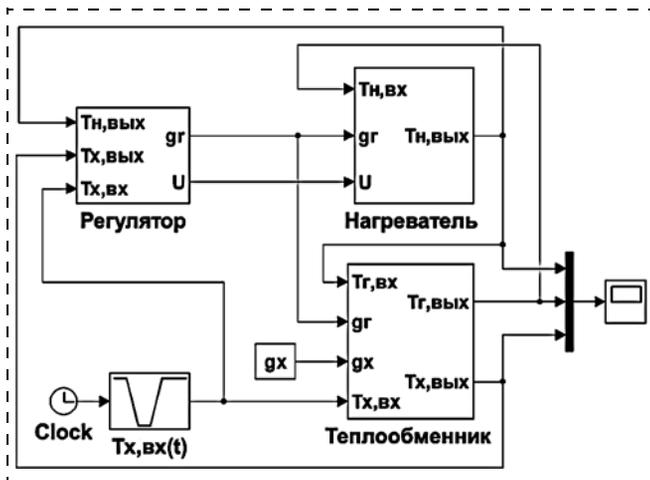


Рис. 11. Имитационная модель системы управления процессом теплообмена

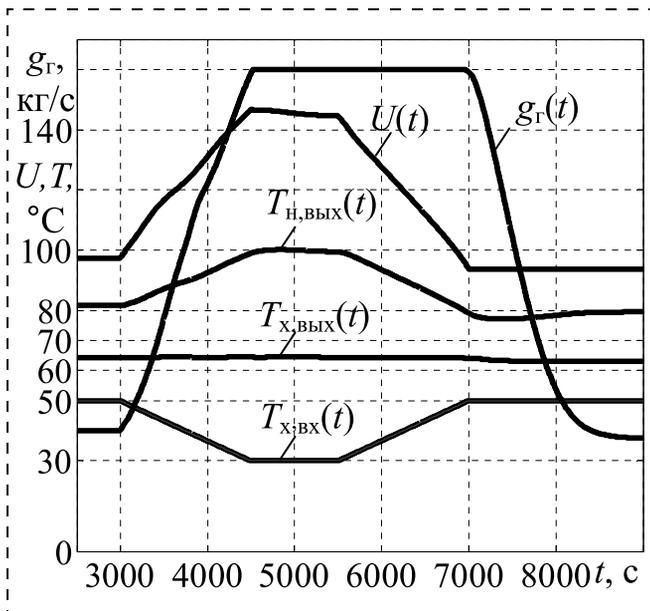


Рис. 12. Переходные процессы в системе с контуром регулирования $T_{n, \text{вых}}$

Необходимость создания этого контура следует из рис. 9 и 10, на которых видно, что значение $T_{n, \text{вых}}$ в процессе регулирования выходит за рекомендуемое максимальное значение $T_{n, \text{вых, макс}} = 100$ °С.

Переходные процессы в системе, объединяющей все три упомянутых выше варианта структуры (рис. 11), показаны на рис. 12.

Из рис. 12 можно заключить, что открытая динамическая система, построенная на основе принципов гомеостатического управления, позволяет полностью обеспечить требуемые показатели качества регулирования рассматриваемого процесса.

Обсуждение результатов

Обсудим результаты, которые были получены благодаря использованию принципов гомеостатического управления:

- многоуровневости обработки информации;
- модульности формирования управляющего решения;
- многообразия методов обработки и разделения функций подсистем.

Использование принципа многоуровневости выразилось в формировании n каналов регулирования по расходу g_r и температуре U с помощью соответствующего числа матриц перехода $M(\gamma_i \times \gamma_j, \alpha_r)$, $i = 1, \dots, n; j = 1, \dots, n; r = 1, \dots, n; n = 5$, формируемых независимо друг от друга, что позволило избавиться от необходимости синтеза единого универсального алгоритма управления, заменив его совокупностью простых локальных процедур, используемых в процессе управления при различных состояниях объекта — регулируемых величин $T_{x, \text{вых}}$ и $T_{n, \text{вых}}$. Взаимосвязь этих каналов управления образует иерархическую структуру по степени отклонения регулируемой величины от заданного значения (для $T_{x, \text{вых}}(t)$) или ее приближения к критическому значению (для $T_{n, \text{вых}}(t)$). Переключение иерархических уровней можно заметить на рис. 10 (см. комментарии к этому рисунку) и рис. 12, оно имеет ясный физический смысл: при нехватке ресурсов низкого уровня управления включается в работу более высокий иерархический уровень системы, обеспечивая в итоге требуемое качество регулирования. При этом переход к более высокому уровню иерархии (за счет соответствующего содержания матриц $M(\gamma_i \times \gamma_j, \alpha_r)$ в разных каналах) сопровождается нелинейным, упреждающим нарастанием компенсирующего управляющего воздействия. Биологическая природа и практическая значимость механизма нарастания компенсирующего воздействия подробно рассмотрена в работе [6].

Модульность формирования управляющего решения в каждом канале системы заключается, как это следует из (10)—(12), в том, что результирующее управление β определяется в результате объединения компонент $\mu_r(\Delta g_r)$ пяти управляющих модулей ($r = 1 \dots 5$), вклад каждой из которых устанавливается текущей мерой ν_r близости искомой

матрицы перехода $M(\gamma \times \gamma_{\text{тр}}, \beta)$ и соответствующих матриц каждого модуля. Использование принципа модульности при построении регулятора позволяет сформировать управление β в виде многомерной функции $\beta(v_1, v_2, v_3, v_4, v_5)$ (в общем случае, в виде n -мерного вектора), обеспечивающей высокую гибкость управления во всем пространстве текущих состояний объекта, в том числе и в нелинейных областях этого пространства.

Принцип многообразия методов обработки информации, в широком его понимании, реализуется, в первую очередь, путем построения в системе нескольких каналов управления с различными алгоритмами принятия решений в каждом канале. Кроме того, в рассматриваемом примере теплового процесса можно указать на введение в структуру системы канала управления по возмущающему воздействию $T_{x, \text{вх}}$ (см. рис. 5 и рис. 11), выступающего в качестве принципиальной альтернативы каналам регулирования по отклонению регулируемых величин. Эффективность введения альтернативного управления подтверждается сравнением качества стабилизации $T_{x, \text{вых}}(t)$ на рис. 9 и рис. 10. Полученный результат хорошо согласуется с применяемым в теории автоматического регулирования принципом двухканальности [19].

Разделение обрабатываемой в системе информации по функциональному признаку (разделение функций подсистем) опирается, как уже отмечалось, на биологические примеры предпочтительности специализированных приспособительных механизмов перед универсальными [9]. В задачах управления многомерными объектами этот принцип реализуется в стремлении обеспечить автономное регулирование выходных координат системы независимыми регуляторами, синтез которых, в силу их автономности, существенно упрощается [20, 21]. В частности, в рассмотренном примере подсистема регулирования температуры теплоносителя $T_{x, \text{вых}}$ вторичного контура функционирует в целях удовлетворения нужд потребителей, а подсистема температуры $T_{n, \text{вых}}$ теплоносителя первичного контура служит для предотвращения аварийных режимов в нагревателе. Как было показано, эти подсистемы реализованы на независимых регуляторах и обеспечивают требуемое качество управления процессом теплообмена одновременно по обеим координатам.

Методологическое единство изложенных выше принципов заключается в формировании и последующем гибком использовании альтернативных вариантов структуры системы управления и алгоритмов ее функционирования. Это единство позволяет объединить их под общим названием принципов многоальтернативности [8], составляющих инструмент практического синтеза систем гомеостатического управления.

На примере синтеза системы регулирования процесса теплообмена продемонстрировано решение задачи технической гомеостатики [6], методы, способы и приемы реализации которой включают в себя, в частности, принципы многоальтернативного управления, позволяющие целенаправленно осуществлять:

- построение многоуровневых регуляторов, в которых алгоритм работы каждого уровня формируется независимо от других уровней управления и включается в работу по мере отклонения системы от требуемого состояния, реализуя в системе свойство гомеостаза;
- модульность компонент результирующих управляющих воздействий. Многообразие значений этих воздействий порождается многообразием текущих вкладов каждой компоненты и обеспечивает гибкость управления в пространстве состояний объекта;
- введение в структуру системы дополнительных, альтернативных каналов ввода внешних воздействий, обеспечивающих требуемое качество регулирования в выбранных режимах работы системы;
- разделение функций регулирования различных координат системы и, как следствие, независимый, автономный синтез соответствующих подсистем управления.

Результаты численного моделирования системы управления тепловым процессом, построенной на основе указанных принципов, указывают на перспективность применения идей гомеостатического управления открытыми динамическими системами с существенной неопределенностью структуры и параметров.

Список литературы

1. Бергаланфи Л. фон. История и статус общей теории систем // Системные исследования. Методологические проблемы. Ежегодник. М.: Наука, 1973. С. 20—37.
2. Эшби У. Р. Конструкция мозга. Происхождение адаптивного поведения. М.: Изд. иностр. лит., 1962. 397 с.
3. Новосельцев В. Н. Теория управления и биосистемы. Анализ сохранительных свойств. М.: Наука, 1978. 320 с.
4. Гомеостатика живых, технических, социальных и экологических систем / Под ред. Ю. М. Горского. Новосибирск: Наука, 1990. 350 с.
5. Филимонов Н. Б. Гомеостатические системы и автомат ограниченный состоянием управляемых динамических объектов // Изв. вузов. Приборостроение. 1998. № 1—2. С. 17—34.
6. Филимонов А. Б., Филимонов Н. Б. Автомат ограничений управляемых динамических процессов // Известия ЮФУ. Технические науки. 2013. № 3 (140). С. 155—161.
7. Филимонов Н. Б. Проблема гомеостатического управления динамическими системами // Устойчивость и процессы управления. Матер. III междунар. конф. СПб.: Изд. дом Федоровой Г. В., 2015. С. 97—98.
8. Подвальный С. Л. Многоальтернативные системы: обзор и классификация // Системы управления и информационные технологии. 2012. Т. 48, № 2. С. 4—13.
9. Подвальный С. Л., Васильев Е. М. Многоальтернативное управление открытыми системами: концепция, состояние и перспективы // Управление большими системами: сборник трудов. М.: ИПУ РАН. 2014. № 48. С. 6—58.

10. Подвальный С. Л., Васильев Е. М. Эволюционные принципы построения интеллектуальных систем многоальтернативного управления // Системы управления и информационные технологии. 2014. Т. 57, № 3. С. 4—8.

11. Александров А. Ю., Жабко А. П., Жабко И. А., Косов А. А. Исследование устойчивости и стабилизация нелинейных переключаемых механических систем на основе декомпозиции // Мехатроника, автоматизация, управление. 2015. Т. 16, № 12. С. 807—812.

12. Косов А. А. О построении стабилизирующего управления для одной механической системы с переключениями // Мехатроника, автоматизация, управление. 2014. № 12. С. 23—26.

13. Подвальный С. Л., Васильев Е. М. Модели многоальтернативного управления и принятия решений в сложных системах // Системы управления и информационные технологии. 2014. Т. 56, № 2.1. С. 169—173.

14. Подвальный С. Л., Васильев Е. М. Интеллектуальные системы многоальтернативного управления: принципы построения и пути реализации // XII Всеросс. совещание по проблемам управления ВСПУ-2014. Институт проблем управления им. В. А. Трапезникова РАН: труды. (Москва, 16—19 июня 2014 г.). М.: ИПУ РАН, 2014. С. 996—1007.

15. Эшби У. Р. Введение в кибернетику. М.: КомКнига, 2005. 432 с.

16. Васильев Е. М. Исследование критических режимов в системах управления теплообменом // Вестник Воронежского государственного технического университета. 2012. Т. 8, № 12.1. С. 69—72.

17. Самарский А. А., Вабищев П. Н. Вычислительная теплопередача. М.: Едиториал УРСС, 2003. 784 с.

18. Деменков Н. П. Нечеткое управление в технических системах: Учебное пособие. М.: Изд-во МГТУ им. Н. Э. Баумана, 2005. 200 с.

19. Петров Б. Н. Избранные труды. Т. 1. М.: Наука, 1983. 429 с.

20. Гайдук А. Р., Плаксиенко Е. А. Синтез автономных и связанных многомерных систем управления // Мехатроника, автоматизация, управление. 2012. № 1. С. 13—20.

21. Филимонов А. Б., Филимонов Н. Б. Метод динамической коррекции и автономизация каналов управления в многосвязных системах на основе формализма линейно-квадратичной оптимизации // Мехатроника, автоматизация, управление. 2012. № 12. С. 2—6.

Homeostatic Control of the Dynamic Systems on the Basis of the Fuzzy Controllers

E. M. Vasiljev, vgtu-aits@yandex.ru✉, A. S. Odnoshivkin, vgtu-aits@yandex.ru
Voronezh State Technical University, Voronezh, 394026, Russian Federation

Corresponding author: Vasiljev E. M., Ph. D., Associate Professor,
Voronezh State Technical University, Voronezh, 394026, Russian Federation
e-mail: vgtu-aits@yandex.ru

Received on April 27, 2016

Accepted on May 12, 2016

The authors present the task of control with account of the significant uncertainty of the dynamic structures and non-stationary parameters of objects. Using the analogy between the task and the ways of functioning of the biological systems, one can conclude that it is possible to solve this problem on the basis of the homeostatic control. This technology combines the principles of a multilevel structure of the information control system, its modular structure and the division of each level. This possibility is demonstrated by the example of control of the process of a double-loop counterflow heat exchange, where the primary loop includes an operated heater and the secondary one is intended for a heat supply of the external end-users. By the temporal and spatial discretization of the process a mathematical model of the heat exchange in a different form was developed and imitated in MatLab. It demonstrates that implementation of those principles is advisable in the class of the fuzzy algorithms, which uses a composition of the local control in each subsystem. The multi-alternative control system of the process of the counterflow heat exchange is synthesized, wherein: we use a controller with five independent channels, which start by the distance of the rated state and ensure the property of homeostasis; an alternative channel of the input disturbance, which operates at a high-speed system, is implemented; an additional control loop of the primary coolant flow, which prevents critical temperature conditions in the heat exchanger is constructed. By a simulation model the quality indicators of the system are checked and the effectiveness of the multi-alternative principles is confirmed. The article ends with a discussion of the concrete practical results, obtained by multilevel processing of information in the system, modular formation of the control solution and variety of processing and separation methods of the subsystem function. Good prospects for use of the multi-alternative concept for the control systems with a fundamental indeterminacy of the structure and the parameters are confirmed.

Keywords: homeostatic control, concept of multi-alternative, hierarchical systems, modularity, separation of functions, fuzzy algorithms, thermal processes

For citation:

Vasiljev E. M., Odnoshivkin A. S. Homeostatic Control of the Dynamic Systems on the Basis of the Fuzzy Controllers, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2016, vol. 17, no. 11, pp. 732—740.

DOI: 10.17587/mau.17.732-740

References

1. Bertalanfi L. fon. *Istorija i status obshhej teorii system* (History and Status of General Systems Theory), *Sistemnye issledovaniya. Metodologicheskie problemy*, Moscow, Nauka, 1973, pp. 20—37 (in Russian).

2. Ashby W. R. *Konstrukcija mozga. Proishozhdenie adaptivnogo povedenija* (Design for a brain. The origin of adaptive behavior), Moscow, Izd. inostr. lit., 1962, 397 p. (in Russian).

3. Novosel'cev V. N. *Teorija upravlenija i biosistemy. Analiz sohranitel'nyh svojstv* (Control theory and biosystems. Analysis preservative properties), Moscow, Nauka, 1978, 320 p. (in Russian).

4. Gorskiy Ju. M. ed. *Gomeostatika zhivyh, tehnicheskikh, social'nyh i jekologicheskikh sistem* (Homeostatics living, technical, social and environmental systems), Novosibirsk, Nauka, 1990, 350 p. (in Russian).

5. Filimonov N. B. *Gomeostaticheskie sistemy i avtomat ograničenijsostojanija upravljajemyh dinamičeskikh ob'ektov* (Homeostatic systems and automatic restrictions state-controlled dynamic objects), *Izv. vuzov. Priborostroenie*, 1998, no. 1—2, pp. 17—34 (in Russian).

6. **Filimonov A. B., Filimonov N. B.** *Avtomat ogranichenij upravljaemyh dinamicheskikh processov* (The automatic machine of restrictions of controlled dynamic processes), *Izvestija JuFU. Tehniceskie nauki*, 2013, no. 3 (140), pp. 155—161 (in Russian).
7. **Filimonov N. B.** *Problema gomeostatsicheskogo upravlenija dinamiche-skimi sistemami* (The problem of homeostatic control of dynamic systems), *Ustojchivost' i processy upravlenija. Proc. of III Internat. Conf.*, SPb., Izd. dom Fedorovoj G. V., 2015, pp. 97—98 (in Russian).
8. **Podvalny S. L.** *Mnogol'ternativnye sistemy: obzor i klassifikacija* (Multi-alternative system: A review and classification), *Sistemy Upravlenija i Informacionnye Tehnologii*, 2012, vol. 48, no. 2, pp. 4—13 (in Russian).
9. **Podvalny S. L., Vasiljev E. M.** A multi-alternative approach to control in open systems: origins, current state, and future prospects, *Automation and Remote Control*, 2015, vol. 76, no. 8, pp. 1471—1499.
10. **Podvalny S. L., Vasiljev E. M.** Evolutionary principles for construction of intellectual systems of multi-alternative control, *Automation and Remote Control*, 2015, vol. 76, no. 2, pp. 311—317.
11. **Aleksandrov A. Ju., Zhabko A. P., Zhabko I. A., Kosov A. A.** Stability Investigation and Stabilization of Nonlinear Switched Mechanical Systems via Decomposition, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2015, vol. 16, no. 12, pp. 807—812 (in Russian).
12. **Kosov A. A.** Design of Stabilizing Control for One Switching Mechanical System, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2014, no. 12, pp. 23—26 (in Russian).
13. **Podvalny S. L., Vasiljev E. M., Barabanov V. F.** Models of Multi-Alternative Control and Decision-Making in Complex System, *Automation and Remote Control*, 2014, vol. 75, no. 10, pp. 1886—1891.
14. **Podvalny S. L., Vasiljev E. M.** *Intellektual'nye sistemy mnogol'ternativnogo upravlenija: principy postroenija i puti realizacii* (Intelligent multialternative control: principles and ways to implement), *XII Vseross. soveshhanie po problemam upravlenija VSPU-2014 Institut problem upravlenija im. V. A. Trapeznikova RAN* (XII All-Russian conference on Control. Institute of Control Sciences of a name V. A. Trapeznikov RAS: Proceedings of the Conference. (Moscow, 16—19 June 2014)), 2014, pp. 996—1007 (in Russian).
15. **Ashby W. R.** *Vvedenie v kibernetiku* (An Introduction to Cybernetics), Moscow, KomKniga, 2005, 432 p. (in Russian).
16. **Vasiljev E. M.** *Issledovanie kriticheskikh rezhimov v sistemah upravlenija teploobmenom* (Research in heat exchange critical regimes control systems), *Vestnik Voronezhskogo Gosudarstvennogo Tehniceskogo Universiteta*, 2012, vol. 8, no. 12.1, pp. 69—72 (In Russian).
17. **Samarskij A. A., Vabishhev P. N.** *Vychislitel'naja teploperedacha* (Computational heat transfer), Moscow, Editorial URSS, 2003, 784 p. (in Russian).
18. **Demenkov N. P.** *Nechjotkoe upravlenie v tehniceskikh sistemah: Uchebnoe posobie* (Fuzzy control in technical systems: Textbook), Moscow, Publishing house of MGTU im. N. Je. Baumana, 2005, 200 p. (in Russian).
19. **Petrov B. N.** *Izbrannye trudy. T. 1* (Selected works. Vol. 1), Moscow, Nauka, 1983, 429 p. (in Russian).
20. **Gajduk A. P., Plaksienko E.** A Synthesis of Separate and Coherent Multivariate Control Systems, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2012, no. 1, P. 13—20 (in Russian).
21. **Filimonov A. B., Filimonov N. B.** Method of Dynamic Correction and Autonomization of Control Channels in Multiply Connected Systems on the Basis of Formalism of Linearly-Square Optimization, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2012, no. 12, pp. 2—6 (in Russian).



29—31 мая 2017 г. в Санкт-Петербурге
на базе АО "Концерн "ЦНИИ Электроприбор" состоится

**XXIV Санкт-Петербургская
МЕЖДУНАРОДНАЯ КОНФЕРЕНЦИЯ
ПО ИНТЕГРИРОВАННЫМ НАВИГАЦИОННЫМ СИСТЕМАМ
(МКИНС 2017)**



Ежегодная конференция стала традиционным местом обмена идеями ученых и инженеров всего мира в области навигации и управления движением.

Председатель программного комитета — Академик РАН проф. В. Г. Пешехонов

Тематика конференции

- Инерциальные датчики, системы навигации и ориентации
- Микромеханические датчики и системы на их основе
- Глобальные навигационные спутниковые системы
- Альтернативные навигационные системы и датчики
- Системы управления, наведения и их элементы
- Интегрированные системы навигации и управления движением

В рамках каждого направления рассматриваются:

схемы построения и конструктивные особенности; методы и алгоритмы;
особенности разработки и применения для различных подвижных объектов и условий движения (аэрокосмические, морские, наземные, подземные);
испытания и метрология.

На конференции не рассматриваются вопросы, затрагивающие военно-техническое сотрудничество, разработки военных технологий и образцов вооружений и военной техники. Программный комитет считает полезным представление обзорных докладов и докладов молодых ученых (до 33 лет).

*Подробную информацию о конференции см. сайте:
<http://www.elektropribor.spb.ru/icins2017/rindex>*

И. А. Каляев, чл.-корр. РАН, kaliaev@mvs.sfedu.ru,

Южный федеральный университет,

А. И. Каляев, канд. техн. наук, anatoly@kalyaev.net,

Я. С. Коровин, канд. техн. наук,

Научно-исследовательский институт многопроцессорных вычислительных систем

Южного федерального университета

Принципы организации и функционирования безлюдного роботизированного производства с децентрализованным диспетчером

Рассматривается проблема создания безлюдных роботизированных производств (БРП), включающих в свой состав некоторое множество роботизированных обрабатывающих центров (РОЦ) различной функциональной направленности. Основу такого БРП должен составлять автоматический диспетчер, функции которого заключаются в построении плана (временного графика) изготовления заказываемого изделия с помощью РОЦ, входящих в состав БРП, а также плана транспортных операций по перемещению комплектующих и заготовок между ними. Организация такого БРП требует решения двух основополагающих задач: во-первых, необходимо разработать "форму" представления задания на изготовление заказываемого изделия, "понятную" для диспетчера БРП, а во-вторых, разработать метод автоматического формирования плана (временного графика) выполнения задания с помощью РОЦ, входящих в состав БРП. Предлагается метод представления задания на изготовление изделия в виде ациклического графа, каждой вершине которого приписана некоторая операция, выполняемая одним из РОЦ. При этом дуги графа определяют операции по транспортировке комплектующих и заготовок между РОЦ. Также предлагается метод диспетчерования БРП с помощью множества программных агентов, физически реализуемых на контроллерах отдельных РОЦ, причем распределение операций задания между РОЦ осуществляется путем их коллективного взаимодействия посредством некоторого информационного канала связи. Разрабатывается подробный алгоритм работы программного агента РОЦ.

Ключевые слова: безлюдное роботизированное производство, автоматический диспетчер, граф задания, программный агент, коллективное взаимодействие, алгоритм работы агента

Введение

Ускорение научно-технического прогресса приводит к усилению конкурентной борьбы на рынке высокотехнологичной продукции. Побеждает тот, кто сможет быстрее довести свою идею до демонстрационного или опытного образца, способного заинтересовать рыночных инвесторов. В этом плане все большую актуальность приобретает проблема создания безлюдных роботизированных производств (БРП), способных в кратчайшие сроки изготавливать разнообразные опытные образцы и единичные изделия по требованиям заказчика [1, 2]. Такое БРП должно включать в свой состав широкий набор роботизированных обрабатывающих центров (РОЦ) различной функциональной направленности. Заказчик формирует и направляет по сети Интернет задания на изготовление своего изделия с помощью РОЦ, входящих в состав БРП. На основании этого задания автоматический диспетчер БРП должен строить план (временной график) изготовления данного изделия, т.е. распределить операции по отдельным РОЦ, а также организовать транспортировку и доставку требуемых комплек-

тующих и заготовок со склада и между отдельными РОЦ. Далее изделие запускается в производство согласно построенному плану.

Организация такого БРП требует решения двух основополагающих проблем: во-первых, необходимо разработать "форму" представления задания на изготовление заказываемого изделия, "понятную" для диспетчера БРП, а во-вторых, разработать метод автоматического формирования плана (временного графика) выполнения задания с помощью РОЦ, входящих в состав БРП. Именно решению этих проблем посвящена настоящая статья.

Формальная постановка задачи

Будем считать, что в состав БРП входит некоторое множество РОЦ R_1, R_2, \dots, R_N , а также два склада — склад комплектующих и заготовок и склад готовых изделий (рис. 1). Все РОЦ и склады объединены общей транспортной линией, посредством которой комплектующие, заготовки и изделия могут передаваться между складами и РОЦ, а также между отдельными РОЦ (рис. 1).

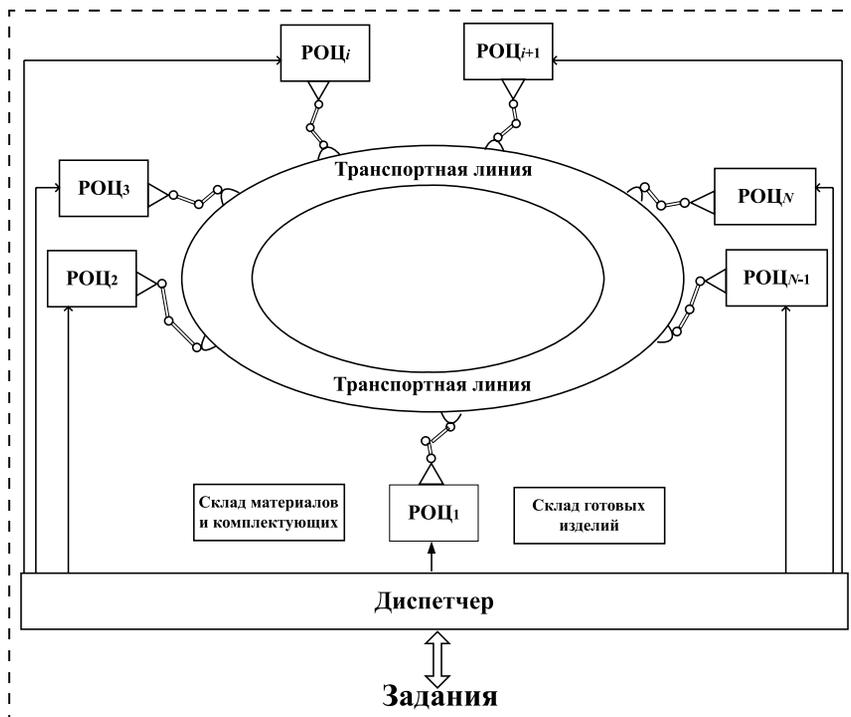


Рис. 1. Структура БРП

Предположим, что каждый РОЦ R_i может выполнять некоторый набор операций $A_i = \langle A_1^i, A_2^i, \dots, A_L^i \rangle$ ($i = 1, 2, \dots, N$), причем в общем случае $A_i \neq A_j$ ($j = 1, 2, \dots, i - 1, i + 1, \dots, N$). Будем считать, что РОЦ R_i выполняет операцию A_l^i ($l = 1, 2, \dots, L$) за время $t_i(A_l^i)$, причем время выполнения идентичных операций различными РОЦ одинаково. Кроме того положим, что время транспортировки комплектующих и заготовок между отдельными РОЦ, а также между РОЦ и складами составляет $t_{\pi}(S)$, где S — длина транспортной линии между ними.

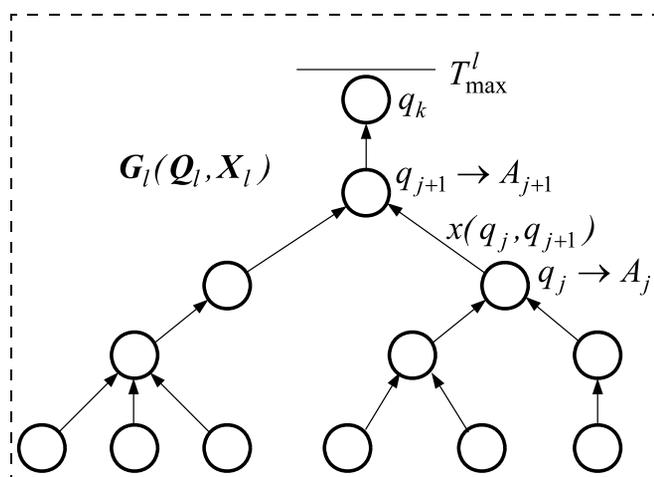


Рис. 2. Граф $G_l(Q_l, X_l)$ задания Z_l

Будем считать, что в БРП в случайные моменты времени через сеть Интернет от различных заказчиков поступает некоторое множество (поток) различных заданий на изготовление изделий $Z = \langle Z_1, Z_2, \dots, Z_M \rangle$, причем для каждого задания Z_l заказчиком устанавливается момент времени T_{\max}^l , к которому он желает получить свое изделие.

Цель работы БРП заключается в выполнении всех заданий на изготовление изделий $Z = \langle Z_1, Z_2, \dots, Z_M \rangle$ к установленным заказчиками моментам времени.

Формализация задания на изготовления изделия

Для того чтобы БРП мог осуществить изготовление некоторого изделия $Z_l \in Z$, задание на его изготовление должно быть представлено заказчиком в некотором унифицированном формализованном виде, понятном автоматическому диспетчеру БРП.

Задание на изготовление некоторого изделия $Z_l \in Z$ можно представить в виде ациклического графа $G_l(Q_l, X_l)$ (рис. 2), каждой вершине $q_j \in Q_l$ которого приписана некоторая операция A_j , принадлежащая множеству операций $A = \bigcup_{i=1}^N A_i$, выполняемых различными РОЦ, причем если две вершины графа q_j и q_{j+1} соединены дугой $x(q_j, q_{j+1})$, то это означает, что операция A_{j+1} , приписанная вершине q_{j+1} , должна выполняться по завершению операции A_j , приписанной вершине q_j . Входные вершины графа $G_l(Q_l, X_l)$ определяют операции по доставке исходных комплектующих и заготовок, необходимых для изготовления изделия, со склада, а конечная вершина определяет операцию по размещению на складе готовой продукции конечного изделия, получаемого в результате выполнения всей программы его изготовления.

Принципы организации диспетчера БРП

После того как задание Z_l на изготовление изделия формализовано в виде графа $G_l(Q_l, X_l)$, оно передается через сеть Интернет в диспетчер БРП, функции которого заключаются в построении плана (временного графика) изготовления изделия, т.е. в распределении отдельных операций задания Z_l между РОЦ с привязкой их выполнения к определенным моментам времени в соответствии с графом задания $G_l(Q_l, X_l)$.

Можно предложить два способа организации такого диспетчера.

В более простом варианте управление БРП может осуществляться с помощью некоторого специально выделенного серверного узла, функции которого заключаются в распределении операций потока поступающих заданий $Z = \langle Z_1, Z_2, \dots, Z_M \rangle$ между РОЦ (см. рис. 1). Однако такая централизованная организация диспетчера БРП имеет целый ряд недостатков. Во-первых, при большом числе РОЦ решение задачи диспетчирования их работы с помощью одного серверного узла будет затруднено вследствие необходимости выполнения большого объема вычислений в реальном времени поступления заданий. Во-вторых, существенно затрудняется масштабирование БРП (т.е. добавление в ее состав новых РОЦ), поскольку при этом необходимо полностью менять не только программу работы центрального диспетчера, но и архитектуру связей диспетчера с РОЦ. И наконец, в-третьих, БРП с центральным диспетчером становится малонадежным, поскольку выход диспетчера из строя приводит к катастрофическим последствиям для всего БРП в целом. Все перечисленные проблемы многократно усложняются в случае, когда БРП должно выполнять не одиночное задание, а поток заданий, поступающих в заранее неизвестные моменты времени.

Всех этих недостатков можно избежать, если использовать принципы децентрализованного мультиагентного диспетчирования в распределенных системах [3—6]. При этом каждый из РОЦ, входящих в состав БРП, должен обладать своим программным агентом, представляющим "интересы" данного РОЦ в процессе диспетчирования, а оптимизация распределения операций поступающих заданий между РОЦ должна осуществляться путем их коллективного взаимодействия посредством некоторого информационного канала связи (рис. 3).

При этом возникает вопрос: каким образом агенты должны взаимодействовать с заказчиками и получать от них задания? Такое взаимодействие может осуществляться посредством некоторого специально пассивного узла, подключенного к сети Интернет и играющего роль "доски объявлений" (ДО), на которой заказчики могут размещать свои задания (рис. 3). При этом дескриптор задания Z_i , размещаемого на ДО, должен содержать:

- граф $G_i(Q_i, X_i)$ задания;
- список вершин множества Q_i и приписанных им операций;
- момент времени T_{\max}^i , к которому потребитель желает получить готовое изделие.

Агенты должны периодически опрашивать ДО с целью загрузить "свой" РОЦ работой. При этом если задание $Z_i \in Z$ не может быть выпол-

нено с помощью одного РОЦ R_i , то агенты должны сформировать некоторую виртуальную организацию — сообщество, состоящее из множества РОЦ $R_i = \langle R_i, R_j, \dots, R_k \rangle$, общей целью которых является выполнение задания $Z_i \in Z$ к установленному заказчиком моменту времени.

В укрупненном виде работу БРП с мультиагентным диспетчером можно представить в следующем виде.

1. Заказчик формирует свое задание $Z_i \in Z$ в виде графа $G_i(Q_i, X_i)$, устанавливает момент времени T_{\max}^i , к которому он хочет получить готовое изделие, и размещает дескриптор задания на ДО.

2. Агент РОЦ R_i , не задействованного в выполнении других заданий, опрашивает ДО в поисках работы для "своего" РОЦ. В случае обнаружения на ДО задания $Z_i \in Z$ агент делает попытку войти в состав сообщества R_i по его выполнению. Для этого агент R_i выделяет фрагмент задания $Z_i \in Z$ (т.е. некоторую последовательность входящих в него операций), не закрепленных ранее за другими агентами, который он может выполнить к установленному моменту времени с помощью "своего" РОЦ. Если такой фрагмент обнаруживается, агент R_i вступает в сообщество R_i и переходит к исполнению принятых на себя операций.

4. Агент R_i контролирует ход выполнения операций принятого к исполнению фрагмента задачи, периодически оценивая время их завершения. В случае если агент по каким-либо причинам не успевает завершить исполнение данных операций к установленному моменту времени, он сообщает об

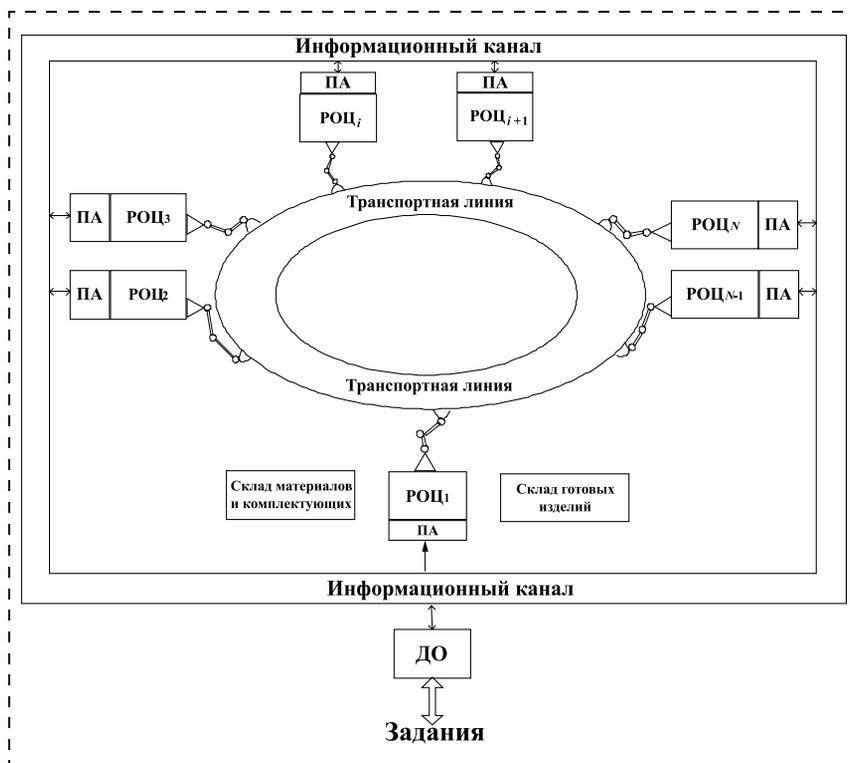


Рис. 3. Структура БРП с мультиагентным диспетчером

этом на доску объявлений и выходит из состава общества R_l .

5. В случае успешного исполнения всех принятых операций задания Z_l агент R_j передает с помощью транспортной линии изделие на другой РОЦ для выполнения последующих операций либо на склад готовой продукции.

Использование принципов децентрализованного мультиагентного диспетчирования БРП обеспечивает: во-первых, высокую отказоустойчивость системы, поскольку в ней отсутствует "узкое горло" в виде центрального диспетчера, а отказ любого из агентов не приводит к катастрофическим последствиям для всего БРП в целом; во-вторых, возможность практически неограниченного увеличения (масштабирования) числа РОЦ в составе БРП путем их простого подключения к информационному каналу связи; и наконец, в-третьих, снижение вычислительной нагрузки на отдельного программного агента при решении задачи диспетчирования, что, в свою очередь, обеспечивает возможность ее решения в реальном времени поступления заданий.

Однако, вместе с тем, использование принципов мультиагентного диспетчирования при управлении БРП требует разработки специального алгоритма работы программного агента РОЦ. Именно разработке такого алгоритма посвящена последующая часть данной статьи.

Алгоритм работы программного агента

Прежде чем приступить к разработке алгоритма мультиагентного диспетчирования БРП, введем понятие "нити". Под нитью будем понимать некоторую последовательность вершин $H_f = \langle q_1^f, q_2^f, \dots, q_k^f \rangle$ графа $G(Q, X)$ задания $Z_l \in Z$, в которой вершины q_j^f и q_{j+1}^f ($j = 1, 2, \dots, k - 1$) соединены дугой

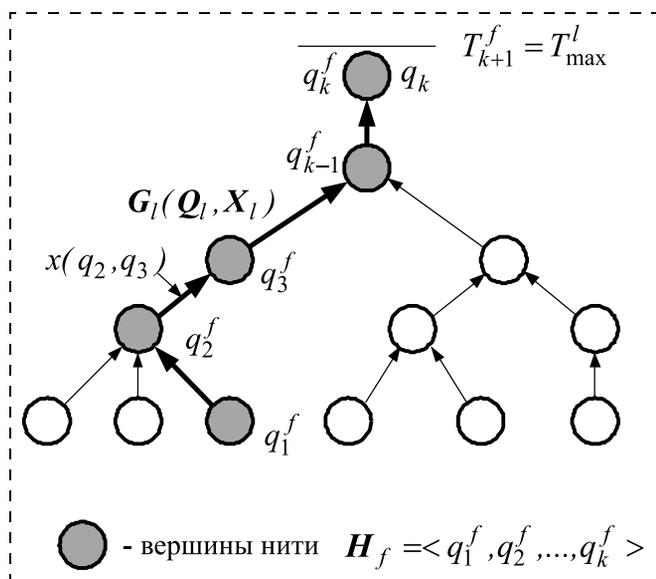


Рис. 4. Выделение нити H_f в графе $G(Q, X)$ задания Z_l

$x(q_j^f, q_{j+1}^f)$ (рис. 4). Иными словами, нить определяет некоторый набор операций задания Z_l , которые должны выполняться последовательно. При этом под длиной t_f нити H_f будем понимать суммарное время, затрачиваемое на выполнение приписанных ее вершинам операций, определяемое как

$$t_f = \sum_{i=1}^k (t_p(A_i^f) + t_{\Pi}(S_p, c)),$$

где $t_p(A_i^f)$ — время, затрачиваемое РОЦ $R_p \in R$ на выполнение операции A_i^f , приписанной вершине $q_i^f \in H_f$ ($i = 1, 2, \dots, k$); $t_{\Pi}(S_p, c)$ — время, затрачиваемое на транспортировку изделия от РОЦ $R_p \in R$, выполняющего операцию A_i^f , к РОЦ $R_c \in R$, выполняющему следующую по очереди операцию A_{i+1}^f нити H_f ; S_p, c — длина транспортной линии между РОЦ R_p и РОЦ R_c .

Если операции A_i^f и A_{i+1}^f выполняются одним и тем же РОЦ R_p , то соответственно $t_{\Pi}(S_p, p) = 0$.

Очевидно, если вся нить H_f выполняется одним ресурсом R_p^f , то ее длина будет составлять

$$t_f = \sum_{i=1}^k t_p(A_i^f). \quad (1)$$

При этом если известен требуемый момент времени T_{k+1}^f исполнения всей нити H_f (рис. 4), то можно определить допустимые моменты времени T_d^f начала выполнения всех операций A_d^f , приписанных ее вершинам $q_d^f \in H_f$ ($d = 1, 2, \dots, k$) (при которых РОЦ R_p успевает выполнить всю нить H_f к требуемому моменту времени T_{k+1}^f), как

$$T_d^f = T_{k+1}^f - \sum_{i=d}^k t_p(A_i^f) \quad (d = 1, 2, \dots, k - 1). \quad (2)$$

Исходя из этих соображений, можно предложить следующую процедуру мультиагентного диспетчирования работы БРП при выполнении потока заданий.

Пользователь формирует свое задание $Z_l \in Z$ в виде графа $G(Q, X)$ и определяет требуемый момент времени T_{\max}^l , к которому ее решение должно быть получено. Дескриптор представленного таким образом задания $Z_l \in Z$ размещается на доске объявлений (см. рис. 2).

Агенты, РОЦ которых не задействованы в выполнении каких-либо заданий, обращаются к ДО в поисках работы. Если агент свободного РОЦ $R_p \in R$ обнаруживает на ДО дескриптор задания Z_b , то он делает попытку войти в сообщество R_l по его исполнению.

Поскольку, как мы приняли выше, каждый РОЦ имеет некоторую специализацию (т.е. может выполнять некоторый ограниченный набор операций), то в общем случае может оказаться, что РОЦ R_p способен выполнять далеко не все операции, приписанные вершинам графа $G_l(Q_b, X_l)$ задания Z_l . Поэтому в графе $G_l(Q_b, X_l)$ необходимо предварительно выделить подграф $G_l^p(Q_b^p, X_l^p)$, вершинам которого приписаны операции множества A_p , выполняемые РОЦ R_p (рис. 5). После этого необходимо проанализировать, есть ли в графе $G_l^p(Q_b^p, X_l^p)$ вершины, для которых установлено требуемое время их исполнения. Отметим, что изначально в момент размещения дескриптора задания Z_l на ДО требуемое время исполнения T_{\max}^l приписано только конечной вершине q_k графа $G_l(Q_b, X_l)$ (см. рис. 2). Если таких вершин нет, то это говорит о том, что агент R_p пока что не может вступить в сообщество R_l по выполнению задания Z_b , и поэтому он вновь переходит к режиму опроса ДО в целях поиска других заданий.

В противном случае агент R_p выделяет в графе $G_l(Q_b, X_l)$, хранящемся в дескрипторе задания Z_l на ДО, наиболее длинную нить $H_1 = \langle q_1^1, q_2^1, \dots, q_k^1 \rangle$ согласно выражению (1), конечной вершине q_k^1 которой приписан требуемый момент времени ее исполнения T_{k+1}^1 (рис. 5). Последнее может быть осуществлено с помощью одного из известных алгоритмов поиска экстремальных путей на графах [7].

Далее агент R_p определяет согласно (2) момент времени T_1^1 , когда ему необходимо приступить к выполнению первой операции нити H_1 , т.е. операции A_1^1 , приписанной вершине $q_1^1 \in H_1$, чтобы успеть завершить исполнение всей нити H_1 к заданному моменту времени T_{k+1}^1 .

Если при этом оказывается, что $T_1^1 < T_{\text{тек}}$, где $T_{\text{тек}}$ — текущий момент времени, то это означает, что РОЦ R_p не может обеспечить выполнение всей последовательности операций нити $H_1 = \langle q_1^1, q_2^1, \dots, q_k^1 \rangle$ к установленному заказчиком моменту времени T_{k+1}^1 . Поскольку мы приняли, что время выполнения однотипных операций различными РОЦ одинаково, то никакой другой РОЦ также не сможет выполнить данную нить к требуемому моменту

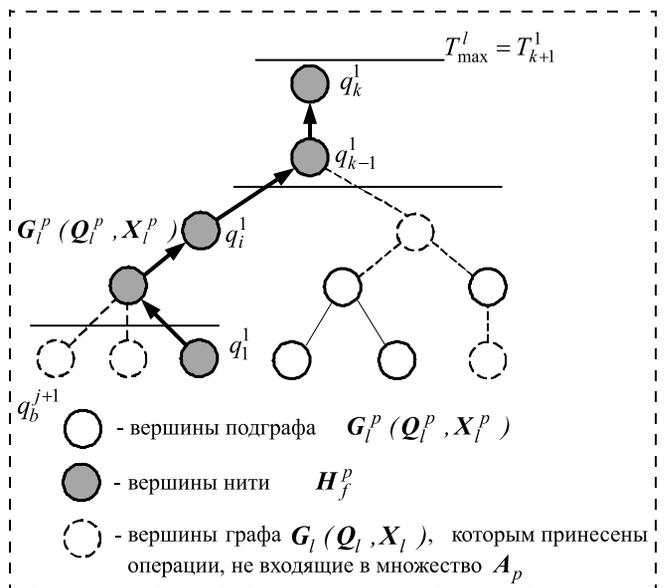


Рис. 5. Выделение подграфа $G_l^p(Q_b^p, X_l^p)$ в графе $G_l(Q_b, X_l)$ задания Z_l

времени. Это говорит о том, что задание Z_l не может быть выполнено к установленному заказчиком моменту времени. В этом случае агент R_p направляет на ДО соответствующее сообщение. Задание Z_l снимается с ДО, а потребителю посылается сообщение о невозможности выполнения его задания к установленному моменту времени. После этого агент РОЦ R_p вновь переходит к опросу ДО в поисках работы.

Если же условие $T_1^1 \geq T_{\text{тек}}$ выполняется для нити H_1 , то в этом случае агент R_p принимает на себя исполнение последовательности операций, приписанных ее вершинам. При этом агент R_p осуществляет модификацию дескриптора задания Z_l на ДО, а именно:

- 1) его идентификатор записывается в список членов сообщества R_l по выполнению задания Z_b ;
- 2) вершины, входящие в нить H_1 , исключаются из графа $G_l(Q_b, X_l)$ задания Z_l , в результате чего формируется новый граф $G_l^1(Q_b^1, X_l^1) = G_l(Q_b, X_l) / H_1$ (рис. 6);
- 3) всем вершинам графа $G_l^1(Q_b^1, X_l^1)$, инцидентным вершинам нити H_1 , приписываются требуемые моменты времени их исполнения, которые определяются исходя из следующих соображений.

Допустим, что некоторая вершина q_f^2 графа $G_l^1(Q_b^1, X_l^1)$ инцидентна вершине q_b^1 , принадлежащей нити H_1 (рис. 6). Это означает, что выполняемая операция, приписанная вершине q_b^1 , может быть начата только по завершении операции, при-

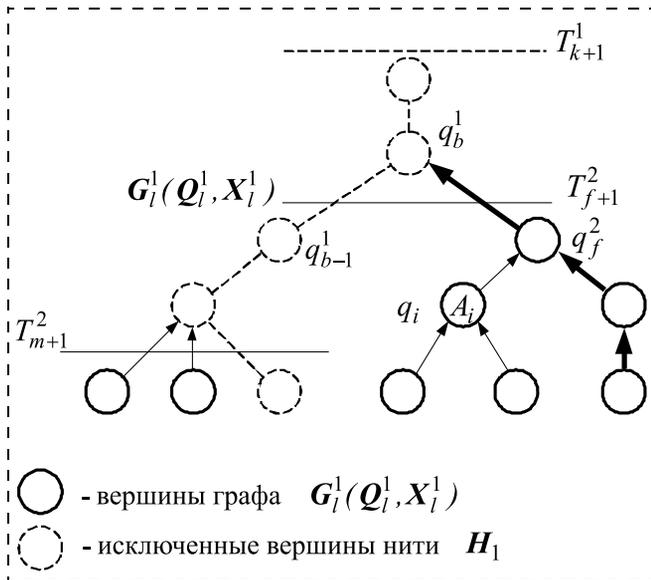


Рис. 6. Граф $G_l^1(Q_l^1, X_l^1)$ задания Z_l , модифицированный агентом R_p

писанной вершине q_f^2 графа $G_l^1(Q_l^1, X_l^1)$. Поэтому, очевидно, что результаты выполнения операции, приписанной вершине q_f^2 , должны быть получены и переданы РОЦ R_p , выполняющему операции нити H_1 , не позже, чем к требуемому моменту времени T_b^1 начала выполнения РОЦ R_p операции A_b^1 , приписанной вершине q_b^1 , определяемому согласно выражению (2) как

$$T_b^1 = T_{k+1}^1 - \sum_{i=b}^k t_p(A_i^1). \quad (3)$$

В противном случае РОЦ R_p не успеет закончить исполнение взятой на себя нити H_1 к требуемому моменту времени T_{k+1}^1 .

Поэтому вершине q_f^2 графа $G_l^1(Q_l^1, X_l^1)$ приписывается требуемое время ее исполнения $T_{f+1}^2 = T_b^1$, а также идентификатор РОЦ R_p , которому результаты исполнения операции, приписанной вершине q_f^2 , должны быть переданы (рис. 6).

Аналогичным образом определяются требуемые моменты T_{m+1}^2 исполнения всех остальных вершин графа $G_l^1(Q_l^1, X_l^1)$, инцидентных вершинам нити H_1 (рис. 6).

Если после модификации новый граф $G_l^1(Q_l^1, X_l^1)$ задания Z_l на ДО еще не пустой, т.е. $G_l^1(Q_l^1, X_l^1) \neq \emptyset$, то процесс создания сообщества R_l для выполнения задания Z_l продолжается далее.

Допустим, что через какое-то время другой свободный РОЦ R_c обнаруживает на ДО дескриптор задания Z_l и предпринимает попытку войти в состав сообщества R_l по его исполнению.

Для этого агент R_c выделяет в графе $G_l^1(Q_l^1, X_l^1)$ подграф $G_l^{1c}(Q_l^{1c}, X_l^{1c})$, вершинам которого приписаны операции, входящие в исполняемое РОЦ R_c множество A_c (рис. 7). Далее в графе $G_l^{1c}(Q_l^{1c}, X_l^{1c})$ агент R_c выделяет наиболее длинную нить $H_2 = \langle q_1^2, q_2^2, \dots, q_f^2 \rangle$, конечной вершине q_f^2 которой приписано требуемое время исполнения T_{f+1}^2 (рис. 7), и анализирует возможность ее исполнения с помощью "его" РОЦ к данному моменту времени. Для этого он с помощью выражения (2) определяет время T_1^2 начала исполнения операции A_1^2 , приписанной первой вершине q_1^2 данной нити H_2 , и сравнивает его с текущим временем $T_{\text{тек}}$. Если $T_{\text{тек}} > T_1^2$, то это означает, что РОЦ R_c не может выполнить данную нить к требуемому моменту времени T_{f+1}^2 , что говорит о том, что все задание в целом также не может быть выполнено к установленному заказчиком моменту времени. В этом случае задание снимается с ДО, заказчику посылается сообщение о невозможности выполнения его задания к установленному времени, а агент R_c снова переходит в режим опроса ДО в поисках работы для "своего" ресурса.

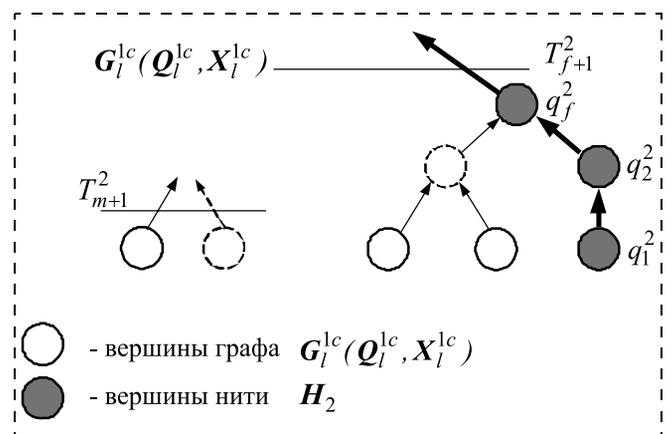


Рис. 7. Выделение подграфа $G_l^{1c}(Q_l^{1c}, X_l^{1c})$ в графе $G_l^1(Q_l^1, X_l^1)$

В случае если условие $T_1^2 \geq T_{\text{тек}}$ выполняется, агент R_c принимает на себя исполнение операций, приписанных вершинам нити H_2 , и осуществляет очередную модификацию дескриптора задания Z_l на ДО:

- идентификатор агента R_c записывается в список членов сообщества R_l по решению задачи Z_l ;
- вершины, входящие в нить H_2 , исключаются из графа $G_l^1(Q_l^1, X_l^1)$, в результате чего образуется новый граф $G_l^2(Q_l^2, X_l^2)$ (рис. 8);
- вершинам q_p^3 графа $G_l^2(Q_l^2, X_l^2)$, инцидентным вершинам q_d^2 нити H_2 , приписывается идентификатор РОЦ R_c , которому результаты исполнения этих операций должны быть направлены, а также требуемое время их исполнения (рис. 8), определяемое как

$$T_{p+1}^3 = T_{f+1}^2 - \sum_{i=d}^f t_c(A_i^2).$$

Далее в процесс распределения операций задания Z_l включается следующий свободный агент, обнаруживший ее дескриптор на ДО, и т.д. до тех пор, пока не окажется, что после очередной модификации граф $G_l^j(Q_l^j, X_l^j)$ стал пустым, что означает, что все операции задания Z_l разобраны агентами, вошедшими в сообщество R_l по его выполнению.

После того как некоторый агент R_p выбрал для исполнения нить $H_f = \langle q_1^f, q_2^f, \dots, q_k^f \rangle$, он приступает к исполнению операций, приписанных ее вершинам с помощью "своего" РОЦ. При этом перед началом выполнения очередной операции A_d^f , приписанной вершине $q_d^f \in H_f$ ($d = 1, 2, \dots, k$), агент R_p должен проверить, во-первых, наличие всех комплектующих и заготовок, необходимых для ее выполнения, а во-вторых, соблюдение временного графика выполнения всей нити H_f в целом.

Поскольку для выполнения подзадачи A_d^f могут потребоваться заготовки, получаемые в результате выполнения смежной нити другим РОЦ R_c , то к моменту начала выполнения РОЦ R_p операции A_d^f может оказаться, что эти заготовки еще не поступили. В этом случае агент R_p должен перейти в режим ожидания поступления необходимых заготовок. Это ожидание может продолжаться до тех пор, пока выполняется условие $T_d^f \geq T_{\text{тек}}$, где T_d^f — требуемое время начала исполнения операции A_d^f ,

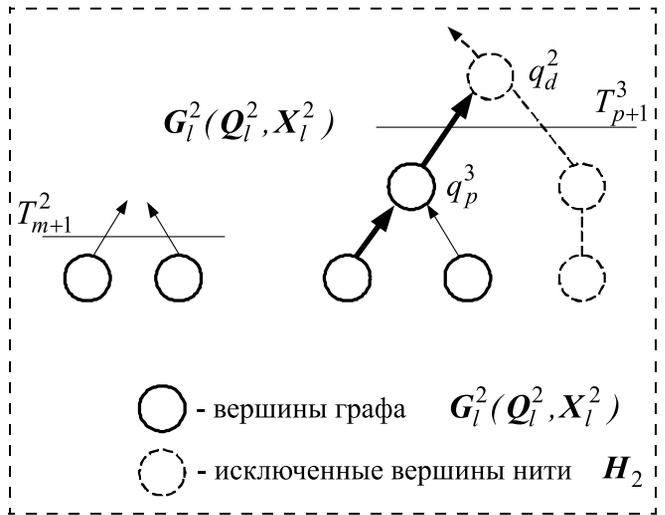


Рис. 8. Граф $G_l^2(Q_l^2, X_l^2)$ модифицированный агентом R_c

приписанной вершине $q_d^f \in H_f$ и определяемое согласно выражению (2).

Если оказывается, что $T_d^f < T_{\text{тек}}$, то это означает, что РОЦ R_p уже не успевает выполнить оставшиеся операции нити H_f к требуемому моменту времени T_{k+1}^f . В этом случае агент R_p должен известить ДО о том, что произошло отставание от графика выполнения задания, и поэтому задание Z_l не может быть решено к установленному заказчиком моменту времени. При этом задание снимается с ДО, пользователю сообщается о невозможности его исполнения к установленному им моменту времени, а всем агентам, вошедшим в сообщество R_l по его решению направляется сообщение о прекращении процесса выполнения задания Z_l , после чего они переходят в режим поиска новых заданий на ДО.

После успешного выполнения всех операций нити H_f агент R_p вновь переходит в режим опроса ДО в целях вхождения в новое сообщество по выполнению следующего задания.

Процесс выполнения задания Z_l продолжается до тех пор, пока не окажется, что список агентов-участников сообщества R_l по его выполнению пуст,

а также пуст граф $G_l^j(Q_l^j, X_l^j)$, хранимый в дескрипторе задания Z_l на ДО. Это означает, что все нити задания успешно выполнены. После этого задание снимается с ДО, а заказчику отправляется сообщение об успешном выполнении его задания.

Описанному выше процессу отвечает следующий укрупненный алгоритм функционирования агента, представляющего РОЦ R_p в процессе диспетчирования работы БРП.

Алгоритм 1

1. Агент свободного РОЦ R_p опрашивает ДО.
 2. При обнаружении на ДО задания Z_l агент R_p анализирует граф задания $G_l^j(Q_l^j, X_l^j)$. Если $G_l^j(Q_l^j, X_l^j) = \emptyset$, то переход к п. 1, иначе

3. В графе $G_l^j(Q_l^j, X_l^j)$ агент R_p выделяется подграф $G_l^{jp}(Q_l^{jp}, X_l^{jp})$, вершинам которого приписаны операции множества A_p , выполняемые РОЦ R_p .

4. Если $G_l^{jp}(Q_l^{jp}, X_l^{jp}) = \emptyset$, то перейти к п. 1, иначе

5. Агент R_p выделяет в графе $G_l^{jp}(Q_l^{jp}, X_l^{jp})$ наиболее длинную нить $H_j^p = \langle q_1^{jp}, q_2^{jp}, \dots, q_k^{jp} \rangle$, конечной вершине которой приписано требуемое время исполнения T_{k+1}^{jp} (в момент размещения задания Z_l на ДО требуемое время $T_{k+1}^{jp} = T_{\max}^l$ приписано только конечной вершине q_k графа $G_l(Q_l, X_l)$). Если таковой нити в графе $G_l^{jp}(Q_l^{jp}, X_l^{jp})$ нет, то перейти к 1, иначе

6. Агент R_p определяет допустимый момент времени, когда необходимо начать выполнение нити $H_j^p = \langle q_1^{jp}, q_2^{jp}, \dots, q_k^{jp} \rangle$, чтобы успеть завершить ее исполнение к требуемому моменту T_{k+1}^{jp} , как

$$T_1^{jp} = T_{k+1}^{jp} - \sum_{i=1}^k t_p(A_i^{jp}).$$

7. Если $T_1^{jp} < T_{\text{тек}}$, где $T_{\text{тек}}$ — текущий момент времени, то переход к п. 16, иначе

8. Агент R_p принимает на себя исполнение нити $H_j^p = \langle q_1^{jp}, q_2^{jp}, \dots, q_k^{jp} \rangle$, для чего модифицирует дескриптор задания Z_l на ДО: записывает в список участников сообщества R_l свой идентификатор; модифицирует граф $G_l^j(Q_l^j, X_l^j)$ задания Z_l путем исключения из него вершин нити $H_j^p = \langle q_1^{jp}, q_2^{jp}, \dots, q_k^{jp} \rangle$, т.е. $G_l^{j+1}(Q_l^{j+1}, X_l^{j+1}) = G_l^j(Q_l^j, X_l^j) / H_j^p$; приписывает вершинам $q_j^{j+1,p}$ модифицированного графа $G_l^{j+1}(Q_l^{j+1}, X_l^{j+1})$, инцидентным вершинам $q_b^{j,p}$ нити $H_j^p = \langle q_1^{jp}, q_2^{jp}, \dots, q_k^{jp} \rangle$, идентификатор РОЦ R_p , которому необходимо передать результаты исполнения операции $A_f^{j+1,p}$, приписанной вершине

$q_f^{j+1,p}$, а также требуемое время их исполнения, определяемое согласно выражению (2).

9. Агент R_p переходит к исполнению последовательности операций, приписанных вершинам нити $H_j^p = \langle q_1^{jp}, q_2^{jp}, \dots, q_k^{jp} \rangle$; $d = 1$.

10. Если $T_d^{jp} < T_{\text{тек}}$, где $T_d^{jp} = T_{k+1}^{jp} - \sum_{i=d}^k t_p(A_i^{jp})$ — требуемое время начала выполнения операции A_d^{jp} , приписанной вершине $q_d^{jp} \in H_j^p$, то перейти к п. 16, иначе

11. Агент R_p проверяет наличие заготовок, необходимых для выполнения операции A_d^{jp} . Если заготовки еще не поступили, то перейти к п. 10, иначе

12. Агент R_p выполняет операцию A_d^{jp} , приписанную вершине $q_d^{jp} \in H_j^p$ с помощью своего РОЦ.

13. Если агенту R_p поступило сообщение о прекращении выполнения задания Z_l , то переход к п. 1, иначе

14. $d = d + 1$, если $d \leq k$, то переход к п. 10, иначе

15. Агент R_p сообщает на ДО об успешном выполнении всех операций, приписанных вершинам нити H_j^p задания Z_l . При этом идентификатор агента R_p исключается из списка членов сообщества R_l по выполнению данного задания. Переход к п. 1.

16. Задание Z_l не может быть выполнено к установленному заказчиком моменту времени T_{\max}^l . Дескриптор задания Z_l удаляется с ДО, заказчику направляется сообщение о невозможности выполнения его задания к требуемому моменту времени, а всем агентам, номера которых записаны в списке участников сообщества R_l по выполнению задания Z_l , передается сообщение о прекращении ее исполнения. Переход к п. 1.

Заключение

В статье описаны обобщенные принципы организации и функционирования безлюдного роботизированного производства с децентрализованным диспетчером. Реализация данных принципов обеспечивает:

- возможности автоматического выполнения потока заданий на производство различных изделий, поступающих от заказчиков в заранее неизвестные моменты времени;
- квазиоптимальное адаптивное распределение ресурсов (РОЦ) БРП с учетом их специализации в реальном времени поступления заданий;
- высокую полезную загрузку работой РОЦ, входящих в состав БРП;
- возможность неограниченного наращивания (масштабируемости) числа различных РОЦ в составе БРП;

- повышенную отказоустойчивость БРП, поскольку в ней отсутствуют узлы, выход из строя которых приводит к выходу из строя всего БРП в целом.

Список литературы

- URL: www.umpro.ru
- URL: www.rosbalt.ru
- Каляев А. И., Каляев И. А.** Метод децентрализованного управления распределенной системой при выполнении потока заданий // Мехатроника, автоматизация, управление. 2015. № 9. С. 585—598.

- Каляев А. И., Каляев И. А., Коровин Я. С.** Метод мультиагентного диспетчирования ресурсов в гетерогенной облачной среде при выполнении потока задач // Вестник компьютерных и информационных технологий. 2015. № 11. С. 34—40.
- Калыев А. И.** Multiagent Approach for Building Distributed Adaptive Computing System // Procedia Computer Science. 2013. Vol. 18. P. 2193—2202 (URL: <http://dx.doi.org/10.1016/j.procs.2013.05.390>)
- Калыев А. И., Коровин Я. С.** Adaptive Multiagent Organization of the Distributed Computations // AASRI Procedia. 2014. Vol. 6. P. 49—58 (URL: <http://dx.doi.org/10.1016/j.aasri.2014.05.008>).
- Рейнгольд Э., Нивергельт Ю., Део Н.** Комбинаторные алгоритмы. Теория и практика. М.: Мир, 1980. 476 с.

Principles of Organization and Operation of an Unmanned Robotic Production with a Decentralized Controller

I. A. Kalyaev, kaliaev@mvs.sfedu.ru✉, Southern Federal University, Taganrog, 344006, Russian Federation,

A. I. Kalyaev, anatoly@kalyaev.net, Institute of Multiprocessing Computing Systems of Southern Federal University, Taganrog, 347928, Russian Federation,

I. S. Korovin, Institute of Multiprocessing Computing Systems of Southern Federal University, Taganrog, 347928, Russian Federation

Corresponding author: Kalyaev I. A., Corresponding Member of the Russian Academy of Sciences, Head of Research, Southern Federal University, Taganrog, 344006, Russian Federation, e-mail: kaliaev@mvs.sfedu.ru

Received on June 07, 2016

Accepted on June 27, 2016

This work is devoted to the topical problem of elaboration of new principles of a decentralized cooperation of the computer systems, which will create a new class of unmanned robotic productions (URP). In the shortest possible time they could produce a variety of prototypes and individual items meeting the remote customers' requirements. Use of the principles of decentralized multiagent scheduling of URP ensures: firstly, high availability of the system, because it lacks the "bottleneck" in the form of a central controller, and a failure of any of the agents does not lead to the catastrophic consequences for the entire URP; secondly, a possibility of virtually unlimited increase of the number of nodes as a part of URP because of an easy connectivity to the information communication channel; and, finally, thirdly, a reduction of the computational load on an individual software agent in the solution of the scheduling tasks, which in turn, makes it possible to get solutions in real time. This article describes the generic principles of organization and functioning of the unmanned robotic productions with a decentralized controller. Implementation of these principles ensures: — a possibility of automation of the approach to the flow of tasks of production of various products coming from customers in advance and in unknown times; — an adaptive quasi-optimal allocation of resources based on their expertise in the real-time of receipt of the assignments; — a high payload operation of the nodes in the URP; — a possibility of unlimited scalability; — an increased fault tolerance, since it lacks components, the failure of which could lead to a failure of the entire system.

Keywords: *unmanned robotic manufacturing, decentralized controller, fault tolerance, multi-agent system, robotic machining center*

For citation:

Kalyaev I. A., Kalyaev A. I., Korovin I. S. Principles of Organization and Operation of an Unmanned Robotic Production with a Decentralized Controller, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2016, vol. 17, no. 11, pp. 741—749.

DOI: [10.17587/mau.17.741-749](https://doi.org/10.17587/mau.17.741-749)

References

- Available at:** www.umpro.ru
- Available at:** www.rosbalt.ru
- Kalyaev A. I., Kalyaev I. A.** *Metod decentralizovannogo upravleniya raspredelennoj sistemoj pri vypolnenii potoka zadaniy* (Decentralized management of distributed system method when the flow

zadaniy), *Mekhatronika, avtomatizaciya, upravlenie*, 2015, no. 9, pp. 585—598 (in Russian).

4. **Kalyaev A. I., Kalyaev I. A., Korovin Y. S.** *Metod mul'tiagentnogo dispetchirovaniya resursov v geterogennoj oblachnoj srede pri vypolnenii potoka zadach* (The method of multi-agent scheduling resources in a heterogeneous cloud environment when the task flow), *Vestnik komp'yuternyh i informacionnyh tekhnologij*, 2015, no. 11, pp. 34—40 (in Russian).

5. **Kalyaev A. I.** Multiagent Approach for Building Distributed Adaptive Computing System, *Procedia Computer Science*, 2013, vol. 18, pp. 2193—2202 (URL: <http://dx.doi.org/10.1016/j.procs.2013.05.390>).

6. **Kalyaev A. I., Korovin Y. S.** Adaptive Multiagent Organization of the Distributed Computations, *AASRI Procedia*, 2014, vol. 6, pp. 49—58 (URL: <http://dx.doi.org/10.1016/j.aasri.2014.05.008>).

7. **Reinhold E., Nivergelt Yu., Deo N.** *Kombinatornye algoritmy. Teoriya i praktika* (Combinatorial algorithms. Theory and practice), Moscow, Mir, 1980, 476 p. (in Russian).

А. Б. Филимонов, д-р техн. наук, проф. filimon_ab@mail.ru,
МГТУ МИРЭА, МАИ (НИУ), г. Москва,
Н. Б. Филимонов, д-р техн. наук, проф., nbfilimonov@mail.ru,
МГУ им. М. В. Ломоносова, ИПУ им. В. А. Трапезникова РАН, г. Москва,
В. Ю. Тихонов, аспирант, bot-32@ya.ru,
МГТУ МИРЭА, г. Москва

Задача прохождения лабиринта интеллектуальными агентами

Рассматривается задача прохождения лабиринта автоматами. Дается краткий обзор работ, посвященных поведению конечных автоматов в шахматных лабиринтах. Обсуждаются особенности постановки задачи обхода лабиринтов интеллектуальными агентами. Анализируется архитектура интеллектуальных агентов, способных распознавать лабиринт и планировать процессы поиска. Предложена формальная схема ситуационного моделирования поведения интеллектуальных агентов. Обсуждаются вопросы оптимизации маршрутов движения в лабиринте.

Ключевые слова: шахматный лабиринт, поведение автоматов, интеллектуальные агенты, архитектура, ситуационная модель поведения, планирование действий, оптимизация маршрутов

"Лабиринты — архитектурные сооружения со сложными коридорами, которые строились для того чтобы приводить в трепет непосвященных".

Мартин Гарднер

Лабиринты как искусственная испытательная среда для экспериментального исследования алгоритмов управления движением автономных мобильных роботов (МР) широко используются в робототехнике. Статья посвящена задачам прохождения лабиринта робототехническими интеллектуальными агентами (ИА).

При разработке агентного подхода к задачам прохождения лабиринтов необходимо учитывать результаты исследований поведения конечных автоматов в лабиринтах (см., например, [1—4]). Они весьма важны, поскольку позволяют понять возможности прохождения лабиринтов реактивными агентами. ИА в отличие от реактивных агентов способны анализировать среду обитания, планировать и оптимизировать свое поведение. В связи с этим для них кардинально меняются и сами постановки и подходы к решению лабиринтных задач.

Физический и виртуальный лабиринты

Существуют различные толкования слова "лабиринт". Для нас наиболее приемлемо определение лабиринта как сооружения со сложной, запутанной системой помещений и ходов.

Классическая задача прохождения лабиринта заключается в следующем. Имеется исполнитель — автономный робот. Он должен пройти из исходной (стартовой) позиции к целевой (финишной). Под *решением лабиринта* подразумевается поиск такого маршрута. Если конфигурация лабиринта (т.е. его карта, план) известна, то должна решаться задача нахождения кратчайшего маршрута. Возможны различные варианты усложнения задачи: поиск определенного объекта (в частности, "клада"); пат-

рулирование в целях обнаружения динамических объектов; необходимость обходить дополнительно выявляемые в ходе движения препятствия и опасные места и т.п.

МР выполняет функции ориентации и движения. Он обладает набором сенсоров, позволяющих определять наличие стены и контролировать расстояние до нее, а также обнаруживать препятствия.

Физические лабиринты применяются в спортивной робототехнике в качестве полигонов для состязаний роботов [5]. Многие проблемы, с которыми приходится сталкиваться создателям спортивных роботов, находятся на переднем крае научных исследований. Убедительным примером этого являются соревнования в формате Micromouse Competition [6] — конкурс маленьких роботов-мышей по поиску пути в центр лабиринта размером 16×16 ячеек. Соревнование состоит из двух этапов. На первом этапе робот исследует лабиринт и пытается построить в своей памяти его карту. На втором этапе он должен достичь центра лабиринта за кратчайшее время. Здесь используются различные алгоритмы поиска, причем к самым популярным относится алгоритм поиска A^* и вариации алгоритма Беллмана.

Условимся далее использовать следующую терминологию: мир лабиринта — это сам лабиринт, расположенные в нем объекты и перемещающиеся по лабиринту роботы; окружающая среда робота — лабиринт и объекты в нем.

В работе [7] отмечается, что важную роль в робототехнике играют программные симуляторы, так как они не только упрощают работу инженеров, но и позволяют исследователям испытывать новейшие алгоритмы искусственного интеллекта и машин-

ного обучения. Виртуальные лабиринты и обитающие в них виртуальные роботы являются программной моделью физического лабиринта и реальных роботов соответственно и используются в качестве компьютерного инструментария исследования и алгоритмизации механизмов поведения МР.

Прямоугольные клетчатые лабиринты

В простейшем случае лабиринт имеет прямоугольную форму и разбит на квадратные клетки (ячейки) одного размера, стороны которых параллельны координатным осям. Все клетки подразделяются на проходимые — белые — и непроходимые (стены) — черные. Из каждой клетки возможно пройти в соседние клетки лишь по горизонтали или вертикали.

Пусть лабиринт имеет форму прямоугольного клеточного поля размера $M \times N$. Тогда карта лабиринта задается в виде двумерного бинарного массива

$$C = [c_{ij}]_{M \times N},$$

причем, если клетка (i, j) является пустой, то $c_{ij} = 0$, а если стеной, то $c_{ij} = 1$. Путь агента может проходить лишь через пустые клетки.

Схему лабиринта можно задать в виде планарного графа

$$L = (V, H), \quad (1)$$

где V и H — множество вершин и множество ребер соответственно. Вершины графа представляют собой проходимые клетки, причем смежные вершины соответствуют соседним клеткам. Естественно полагать, что граф (1) является связным и без мостов.

Задача о лабиринте формулируется как поиск маршрута, соединяющего две заданные вершины $v_A \in V$ и $v_B \in V$.

Виртуальный робот (далее просто — робот) выполняет функции навигации и движения. Он за один ход может переместиться в соседнюю свободную клетку в одном из четырех возможных направлений: север (n), восток (o), юг (s), запад (w). Карта лабиринта не известна роботу до начала движения.

Робот оснащен набором локационных сенсоров, позволяющих обнаруживать объекты вокруг. К примеру, это может быть датчик касания, определяющий наличие стены спереди. В более сложном варианте — это датчики обзора окружающего пространства (сканеры, дальномеры) с ограниченным радиусом действия.

Если робот начнет движение в сторону находящейся рядом с ним стены, то он разрушится. В процессе движения робот осуществляет контроль стен: анализирует наличие стены, и если стена есть (сработал датчик касания), то движение робота к стене блокируется.

Поведение автоматов в лабиринтах

Важным направлением в теории автоматов является изучение поведения автоматов в лабиринтах. Оно развивалось в основном благодаря усилиям ученых Германии, России, Сербии и Черногории, США и Японии. Кратко обсудим основные понятия, проблематику и достижения в данной области исследований, основываясь на материале работ В. Б. Кудрявцева, Г. Килибарды и Ш. Ушчумлича [1–4].

Заметим, что первый самообучающийся МР — изобретенная К. Шенноном [8] робот-мышь — умел "самостоятельно" находить дорогу в лабиринте, используя алгоритм Люка—Тремо [9].

В теоретических исследованиях прохождения лабиринта автоматами используется понятие шахматного лабиринта, который далее условимся рассматривать как геометрический граф [10].

Шахматный лабиринт

Далее используются стандартные обозначения: \mathbf{R} , \mathbf{Z} , \mathbf{Z}_+ и \mathbf{N} — поле вещественных чисел, множество целых чисел, множество неотрицательных целых и натуральных чисел соответственно.

Множество отрезков \mathbf{T} в \mathbf{R}^2 называется *регулярной конфигурацией отрезков*, если любые два отрезка из этого множества могут иметь не более одной общей точки, причем, если она есть, то обязательно является концевой для обоих отрезков. Пример такого множества дает рис. 1.

Лабиринт (1), где $V \subset \mathbf{R}^2$, называется *прямоугольным плоским лабиринтом*, если

- 1) все ребра являются отрезками, параллельными координатным осям;
- 2) множество ребер является регулярной конфигурацией отрезков.

Ребра лабиринта отмечаются направлением перехода к смежным вершинам (n, o, s, w), а маршруты задаются словами в цепном коде Фримена [11].

Если $V \subseteq \mathbf{Z}^2$, то такой лабиринт называют *целочисленным*, а если при этом длина каждого ребра равна 1, то *мозаичным*.

Пусть L — некоторый прямоугольный плоский лабиринт. Множество $\mathbf{R}^2 \setminus L$ является открытым и в общем случае несвязным. Каждую из компонент

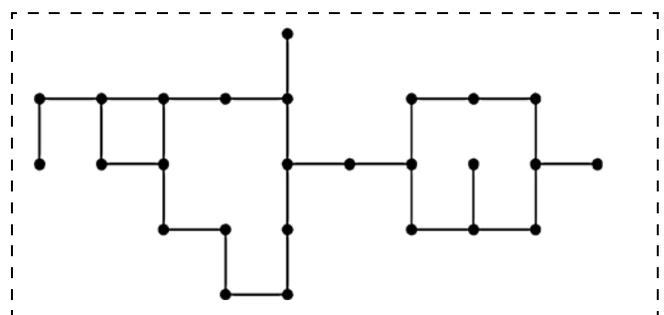


Рис. 1

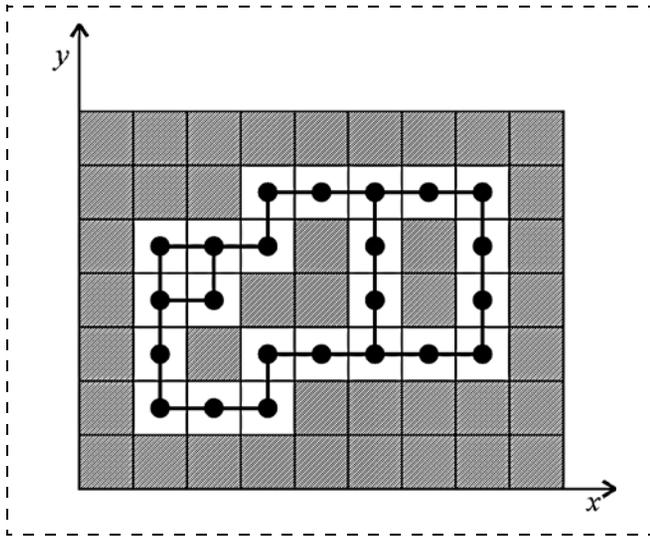


Рис. 2

связности этого множества называют дырой (преградой) лабиринта. Дыра может быть ограниченной или неограниченной. Лабиринт называют $(k + 1)$ -связным, если у него $k \in \mathbf{Z}_+$ ограниченных дыр. Например, мозаичный лабиринт, изображенный на рис. 1, является четырехсвязным.

Пусть L — некоторый целочисленный плоский лабиринт. Любое множество вида $\Delta \cap \mathbf{Z}^2$, где Δ — дыра лабиринта L , назовем \mathbf{Z} -дырой или целочисленной дырой данного лабиринта. Она может быть ограниченной и неограниченной.

К примеру, у лабиринта на рис. 1 две \mathbf{Z} -дыры, а у лабиринта на рис. 2 — три \mathbf{Z} -дыры.

Шахматный лабиринт — это мозаичный лабиринт, у которого любые две вершины, расстояние между которыми равно единице, являются смежными. На рис. 2 дан пример плоского шахматного лабиринта.

Лабиринтные машины

В лабиринты можно "помещать" разного рода математические машины. Самыми простыми из них являются конечные автоматы. Приведем основные понятия, относящиеся к лабиринтным машинам, на основе материалов работ [1—4].

Лабиринт L с выделенной вершиной $v_0 \in V$, называемой начальной, будем считать *инициальным* и обозначать L_{v_0} .

Обозначим $\mathcal{L}(\Omega, \Sigma)$ класс всех лабиринтов с множеством отметок вершин Ω и множеством отметок ребер Σ .

Пусть Ω и Σ — непересекающиеся алфавиты букв ω и σ , которые берутся для отметок вершин и ребер соответственно.

Автомат \mathcal{A} определяется пятеркой

$$\mathcal{A} = (A, Q, B, \varphi, \psi),$$

где A, Q и B — конечные алфавиты — соответственно, входной, состояний и выходной; $\varphi: Q \times A \rightarrow Q$ и $\psi: Q \times A \rightarrow B$ — функции переходов и выходов соответственно. При фиксации начального состояния $q_0 \in Q$ имеем инициальный автомат \mathcal{A}_{q_0} .

Автомат может передвигаться в лабиринте исходя из доступной ему (т.е. находящейся в поле зрения) локальной информации. Возможны различные формы поля зрения. Например, это может быть квадрат 3×3 с центром в текущей вершине.

Пусть A^* и B^* — множества всех слов над алфавитами A и B соответственно. Под функционированием автомата \mathcal{A}_{q_0} понимаем отображение

$$F(\mathcal{A}_{q_0}): A^* \rightarrow B^*,$$

определяемое рекуррентно:

$$q(1) = q_0, \quad (2)$$

$$q(t + 1) = \varphi(q(t), a(t)), \quad (3)$$

$$b(t) = \psi(q(t), a(t)), \quad (4)$$

где $q_0, q(t) \in Q, a(t) \in A, b(t) \in B$.

Автомат \mathcal{A} называется *допустимым* для класса лабиринтов $\mathcal{L}(\Omega, \Sigma)$, если его входной алфавит состоит из букв a вида $(\omega, \{\sigma_1, \dots, \sigma_m\})$, где $\omega \in \Omega, \{\sigma_1, \dots, \sigma_m\} \subseteq \Sigma$, а выходной алфавит есть $\Sigma \cup \{\theta\}$, где $\theta \notin \Sigma$ — некоторый фиксированный символ, смысл которого будет разъяснен ниже. При этом всегда $\psi(q, a) \in \{\sigma_1, \dots, \sigma_m\} \cup \{\theta\}$. Обозначим класс всех таких автоматов $\text{Aut}(\Omega, \Sigma)$.

Пусть \mathcal{A}_{q_0} — некоторый инициальный автомат, а L_{q_0} — некоторый инициальный лабиринт. Интерпретируем функционирование автомата \mathcal{A}_{q_0} в лабиринте L_{q_0} следующим образом. Автомат \mathcal{A}_{q_0} помещается в начальный момент в вершину v_0 лабиринта L_{v_0} .

Предположим, что в какой-то момент автомат оказался в вершине v лабиринта и находится в состоянии q . Он обзрывает звезду, образованную ребрами, инцидентными вершине v . Входной буквой автомата в этот момент является пара, образованная отметкой вершины v и множеством отметок этой звезды. В следующий момент времени получим: если $\psi(q, a) = \theta$, то автомат остается на месте, а если $\psi(q, a) \neq \theta$, то переходит в новое положение — вершину лабиринта, в которое ведет ребро с отметкой $\psi(q, a)$.

Поведением автомата \mathcal{A}_{q_0} в лабиринте L_{v_0} назовем последовательность пар

$$\pi(\mathcal{A}_{q_0}, L_{v_0}) = (q_0, v_0)(q_1, v_1)\dots,$$

определяемых уравнениями (2)—(4).

Будем говорить, что автомат \mathcal{A}_{q_0} *обходит вершину лабиринта* $u \in V(L_{q_0})$, если существует $q \in Q(\mathcal{A}_{q_0})$ такое, что $(q, u) \in \pi(\mathcal{A}_{q_0}, L_{q_0})$.

Обозначим $\text{Int}(\mathcal{A}_{q_0}, L_{q_0})$ множество всех вершин, которые обходит данный автомат. Очевидно, что

$$\text{Int}(\mathcal{A}_{q_0}, L_{q_0}) = \bigcup_{i=1}^{\infty} \{v_i\}.$$

Пусть $L_{q_0} \in \mathcal{L}(\Omega, \Sigma)$ и $\mathcal{A}_{q_0} \in \text{Aut}(\Omega, \Sigma)$. Будем говорить, что автомат \mathcal{A}_{q_0} *обходит лабиринт* L_{q_0} , если

$$\text{Int}(\mathcal{A}_{q_0}, L_{q_0}) = V(L_{q_0}).$$

В противном случае лабиринт L_{q_0} именуется *ловушкой* для автомата \mathcal{A}_{q_0} .

Эти понятия можно расширить для любых сочетаний инициальных или неинициальных автоматов и лабиринтов. Понятия "сильно обходит" и "сильная ловушка" применимы в том случае, если эти свойства имеют место для всех вершин лабиринта и всех состояний автомата.

Наряду с поведением автомата в лабиринте можно также рассматривать поведение системы автоматов в лабиринте. В этом случае понадобится более общая трактовка инициального лабиринта — в нем выделено несколько вершин: L_{v_1, \dots, v_n} .

Пусть задана система допустимых автоматов \mathcal{A} для класса лабиринтов $\mathcal{L}(\Omega, \Sigma)$:

$$\mathcal{A} = (\mathcal{A}_{q_1}^1, \dots, \mathcal{A}_{q_n}^n).$$

Далее L_{v_1, \dots, v_n} — инициальный лабиринт с выделенными вершинами $v_1, \dots, v_n \in V(L)$.

Поведением независимой системы автоматов \mathcal{A} в лабиринте L_{v_1, \dots, v_n} является упорядоченный набор поведений отдельных автоматов:

$$(\pi(\mathcal{A}_{q_1}^1, L_{v_1}), \dots, (\pi(\mathcal{A}_{q_n}^n, L_{v_n}))).$$

Будем говорить, что система автоматов \mathcal{A}

- *обходит лабиринт* L_{v_1, \dots, v_n} , если для некоторого $i \in 1: n$

$$\text{Int}(\mathcal{A}_{q_i}^i, L_{q_i}) = V;$$

- *A-обходит лабиринт* L_{v_1, \dots, v_n} , если

$$\bigcup_{i=1}^n \text{Int}(\mathcal{A}_{q_i}^i, L_{q_i}) = V.$$

В противном случае лабиринт L_{v_1, \dots, v_n} называется *ловушкой* и, соответственно, **A-ловушкой** для \mathcal{A} .

Лабиринт L называется *сильной ловушкой* для системы автоматов \mathcal{A} , если для любых $v_1, \dots, v_n \in V(L)$ лабиринт L_{v_1, \dots, v_n} является ловушкой для \mathcal{A} .

Система \mathcal{A} называется *коллективом*, если входной алфавит каждого автомата системы включает коды текущих состояний всех других автоматов.

Проблема обхода лабиринта автоматами

Активное изучение поведения автоматов в лабиринтах и на графах началось после появления работ К. Делпа [12, 13]. В них была формализована модель К. Шеннона: в качестве лабиринта рассматривалась подобная шахматной доске связанная конфигурация клеток на плоскости или аналогичных кубиков в пространстве (шахматные лабиринты), а в качестве автоматов — конечные автоматы, которые, обозревая некоторую окрестность клетки, в которой они находятся, могут перемещаться в соседнюю клетку в одном из координатных направлений. Выделен, как актуальный, вопрос о существовании автомата, обходящего все такие лабиринты.

Л. Будах [14] показал, что множество всех односвязных инициальных плоских шахматных лабиринтов может быть обойдено одним инициальным автоматом с некоторым фиксированным числом состояний. Им также было высказано предположение о существовании шахматного лабиринта-ловушки на плоскости. Х. Мюллер [15] показал, что всегда в качестве ловушки можно выбрать шахматный лабиринт, у которого не больше трех дыр.

Исходя из этого в работе [1] делается вывод об ограниченности аналитических возможностей конечных автоматов, что в определенном смысле характеризует их негативно. Вместе с тем, интерес представляет следующий вопрос: для каких содержательно интересных классов лабиринтов существуют обходящие их системы?

А. Н. Зыричев [16] установил, что для любого $n \in \mathbb{N}$ класс всех плоских шахматных лабиринтов, имеющих дыры с диаметром не более n , обходится одним автоматом. Показано, что для этого класса существует универсальный обходчик.

Невозможность обхода всех плоских шахматных лабиринтов одним автоматом привела к необходимости рассмотрения вопроса об усилениях модели конечного автомата, решающих задачу обхода. Возможны различные варианты таких усилений.

Одно такое направление усиления модели автомата, предложенное Р. Фишером [17], состоит в разрешении автомату делать отметки в лабиринте. Данная возможность, по существу, означает, что автомат обладает внешней неограниченной памятью.

Главные же варианты усиления возможностей одного автомата связаны с использованием камер, а также переходом к системе автоматов.

Камень — это флажок, маркер, который автомат может забирать или оставлять в вершине графа, общее число камней является постоянным. Фактически камни играют роль ограниченной внешней памяти автомата.

Х. Антельман, Л. Будах и Х.-А. Роллик [18] показали, что для любой конечной системы невзаимодействующих автоматов можно построить ловушку, а также построили бесконечную ловушку сразу для всех допустимых автоматов.

Возможности коллективов автоматов при обходе лабиринтов много шире, чем возможности независимых систем автоматов. Коллектив автоматов в отличие от независимой системы автоматов анализирует лабиринты с учетом положения его членов в лабиринте.

М. Блюм и Д. Козен [19] показали, что один автомат с двумя камнями или коллектив из двух автоматов могут обойти все конечные плоские мозаичные лабиринты. Ф. Хофман [20] установил, что один автомат с одним камнем не решает эту задачу.

В работе [4] дается обзор работ по проблеме распознавания шахматного лабиринта с помощью коллектива автоматов. Обсуждаемая проблемная область охватывает следующий круг вопросов:

1) возможность обхода коллективом автоматов лабиринтов различных классов;

2) задачу о встрече коллективов автоматов в лабиринте;

3) взаимодействие двух автоматов, имитирующих поведение "хищник—жертва", где "хищник" пытается догнать жертву, а та — убежать от него.

Возможны также различные варианты усложнения задачи прохождения лабиринта. Одно из них заключается в размещении препятствий случайным образом в коридорах лабиринта. Положим, что автомат знает карту лабиринта, но расположение препятствий ему не известно. Тогда препятствие остановится обнаруженным автоматом только в зоне его прямой видимости.

С задачей обхода тесно связана проблема остановки автомата в лабиринте после его обхода.

Подытожим изложенное, следуя работе [4].

Изучение поведения автоматов в лабиринтах привело к созданию и утверждению важного направления в теории автоматов. Разработан понятийный аппарат, позволяющий решать имеющиеся проблемы и формулировать новые.

Основная проблематика для автоматов в лабиринтах включает задачу описания тех автоматов и коллективов автоматов, которые обходят лабиринты из заданного класса. В случае, когда для класса лабиринтов отсутствуют автоматы заданного типа, обходящие эти лабиринты, возникает задача выделения ловушек. В качестве главной модели выступают конечные автоматы в конечных и бесконечных плоских мозаичных лабиринтах.

В работе [21] рассматриваются задачи управления распределенными мобильными системами, элементами которых являются группа МР и, возможно,

совокупность стационарных сенсорных устройств, предназначенных для мониторинга окружающей среды. Движение данной системы задается в *терраине* — ограниченной прямоугольной рамкой области с препятствиями. Для точек террайна вводится отношение видимости: две точки x, y видимы одна из другой (что записывается как $x \sim y$), если отрезок $[x, y]$ не пересекает препятствий (но может их касаться). Это отношение является толерантностью, т.е. удовлетворяет свойствам рефлексивности и симметричности (но не обязательно является транзитивным).

В работе [21] в качестве ведущей выделяется задача автономного мобильного патрулирования — информационного мониторинга среды, одной из целей которого является обнаружение и сопровождение динамических объектов. Данный тип задач берет начало от задачи Парсонса "преследования—уклонения" на графах, сформулированной им в 1976 г. [22, 23]. В работе [21] авторы приводят обзор своих работ и работ других авторов по рассматриваемой проблематике.

В работе [24] отмечается, что при распознавании графа несколькими блуждающими по нему агентами основная проблема состоит в обеспечении эффективности их взаимодействия в целях уменьшения затрат времени и памяти на распознавание. Для этого необходимо, чтобы блуждающие агенты не мешали друг другу и не дублировали работу друг друга. Необходимый результат предлагается достичь посредством окраски элементов графа, а также информационного взаимодействия агентов.

В работе [25] рассматривается задача построения автономным мобильным агентом топологической модели своей операционной среды, которая представляет собой связный неориентированный граф с помеченными вершинами. Автором предложен полиномиальный алгоритм восстановления и разметки графа среды для коллектива из агента-исполнителя и агента-вычислителя.

Мультиагентные системы

В общем арсенале средств интеллектуализации автоматических систем все большее признание получает агентная парадигма [26, 27], которая во многом определяет перспективы развития интеллектуальной робототехники [28].

Под *агентом* понимается любая сущность, которая может воспринимать среду обитания (внешний мир) и воздействовать на нее [29—31]. Искусственный агент может иметь программно-аппаратную (робот) или программную (софбот) реализацию. Программные агенты — это компьютерные программы, которые исполняются асинхронно в соответствии с предписанным им целенаправленным поведением.

Агент способен автономно решать возложенные на него задачи, адаптироваться к изменениям во

внешнем окружении, а также общаться с другими агентами для достижения глобальных целей.

Агенты можно классифицировать по степени развития внутреннего представления внешнего мира и способу реализации поведения. Исходя из этого выделяют два типа агентов — реактивные и интеллектуальные.

Реактивные агенты имеют примитивную внутреннюю модель внешнего мира. Для них характерно использование концепции состояния и простейших механизмов поведения типа "стимул—состояние—реакция". Как следствие, реактивные агенты обладают весьма ограниченной возможностью предвидения и практически не способны планировать свои действия. В простейшем варианте реакция агента на внешние события генерируется конечным автоматом, а их поведение описывается языком формальных грамматик.

ИА отличаются от реактивных агентов наличием у них встроенной *базы знаний* и развитого *механизма планирования действий*.

Ментальные свойства агента выражают следующие категории:

- *ощущения* (perceptions) — восприятие обстановки;
- *убеждения* (beliefs) — переменная часть знаний агента о мире;
- *цели* (goals) — желаемый результат воздействий на внешний мир;
- *намерения* (intentions) — сформировавшийся план действий.

Многоагентная система (МАС) — система агентов, взаимодействующих между собой. МАС строится как сообщество агентов, которые могут общаться друг с другом с помощью некоторого языка коммуникаций ACL (Agent Communication Language). Агрегированное поведение МАС проявляется вследствие локальных взаимодействий отдельных агентов, которые осуществляют посредством коммуникаций. У агентов общая цель. Предполагается их взаимодействие для ее достижения.

Поведение агента определяется процессами выполнения базовых функций. Принципы организации этих процессов и структура их информационного взаимодействия определяются архитектурой ИА.

Наибольшее признание получили гибридные архитектуры ИА [32, 33], в которых объединяются реактивный уровень, поддерживающий способность агента быстро реагировать на события, и делиберативный уровень, обеспечивающий решение задач планирования.

Функциональность ИА основана на решении следующих задач:

- восприятие обстановки, поддержка коммуникаций и исполнение действий;
- контроль внешней обстановки и результатов действий агентов;
- реактивное управление поведением;
- локальное планирование поведения;
- организация кооперативного поведения;

- анализ окружающей среды в целях построения ее модели;
 - прогнозирование состояния внешнего мира;
 - стратегическое планирование поведения и др.
- Данные задачи распределяются по иерархическим уровням вертикально организованной архитектуры агентов [32].

Наглядным примером может служить InteRRaP-архитектура — ее управляющий компонент имеет три иерархических уровня: реактивный, локально- и кооперативного планирования [34, 35].

Реактивный уровень реализует реактивное поведение агента, причем для этого используются фрагменты поведения — шаблонные заготовки реакции агента на стандартные ситуации.

Уровень локального планирования позволяет строить локальные планы автономного поведения агента и активирует реактивное поведение через нижележащий уровень управления.

Уровень кооперативного планирования конструирует планы коллективного поведения агентов.

Информационная структура агента включает *рабочую память*, которая служит для хранения текущих данных.

Основу компетенции агента составляют процедурные и декларативные знания о мире: они хранятся в иерархической базе знаний и необходимы для решения функциональных задач.

Общая для всех агентов оперативная информация отображается на доске объявлений.

Под *миссией* агента будем понимать главную задачу (главную цель) его деятельности. Конкретизация миссии приводит к *дереву целей*, которое состоит из целей нескольких уровней: главная цель и подчиненные ей цели первого, второго и последующего уровней. Самый нижний уровень образован локальными целями. Достижение главной цели предполагает реализацию частных целей.

Высокоуровневой моделью поведения агентов является *сценарий деятельности* — он предписывает правила поведения агентов для достижения локальных целей.

Нас интересует вопрос применения ИА в качестве универсальных обходчиков лабиринтов. Поэтому их архитектура должна позволять решать следующие базисные задачи:

- 1) составить карту лабиринта;
- 2) спланировать оптимальный маршрут движения к цели;
- 3) исполнить движение по выработанному маршруту;
- 4) реализовать обход дополнительно выявляемых в ходе движения препятствий и опасных мест.

При этом необходимо учитывать возможность их кооперирования для достижения общих целей.

Пример. Рассмотрим задачу обхода шахматного лабиринта группой агентов. Положим, зоной обзора агента является окрестность фон Неймана, т.е. четыре его окружающие клетки. Двигательный ме-

ханизм агента дает возможность делать шаг в пределах данной окрестности.

Здесь частные цели — обход каждого помещения, составление плана помещения, составление карты лабиринта, прокладка кратчайших маршрутов и др.

Сценарий устанавливает общие правила действий агентов наподобие следующих:

1. Помещения обходятся последовательно одно за другим.

2. Каждое помещение вначале проходится по его периметру, в результате чего устанавливается наличие проходов в нем.

3. Если в неразведанное или частично разведанное помещение зашли несколько агентов, то один из них продолжает его обход, а остальные стремятся покинуть его.

4. План каждого разведанного помещения наносится на карту лабиринта.

5. Для прохождения к неразведанным помещениям через разведанную часть лабиринта прокладываются кратчайшие маршруты и т.п.

Мир Вампуса

Простой испытательной средой для ИА являются микромиры [36], одним из которых является так называемый мир Вампуса [31]. Виртуальные лабиринты по существу являются такими же микромирами.

Мир Вампуса — это пещера, состоящая из залов, соединенных проходами. По пещере бродит Вампус — страшный зверь, который поедает каждого, кто входит в зал, где он находится. В пещере действует агент (охотник), он ищет золото. Агент может убить Вампуса стрелой. В пещере имеются бездонные ямы, в которые может провалиться агент, но это исключено для Вампуса, поскольку он слишком велик для них. Местоположение ям, золота и Вампуса неизвестны.

Уточним исходные данные задачи.

Среда. Пещера моделируется в виде клеточного пространства, в котором залы представлены клетками. Проходам между залами отвечают общие границы клеток. Агент не знает конфигурацию среды.

Исполнительные механизмы. Агент может двигаться вперед, поворачиваться влево и вправо на 90°. Он может выстрелить стрелой лишь один раз и прямо перед собой.

Датчики. Агент имеет пять датчиков, посредством которых он воспринимает внешний мир: 1) чувствует ветерок в соседних с ямой залах; 2) попадая в зал с золотом, видит его; 3) ощущает соприкосновение со стеной; 4) чувствует неприятный запах, если Вампус находится в соседнем зале; 5) слышит жалобный крик смертельно пораженного Вампуса.

Интеллект. Поведение агента базируется на логических рассуждениях, для чего применяются пропозициональная логика и логика предикатов.

Далее при обсуждении функциональных свойств ИА будут использоваться примеры, относящиеся к миру Вампуса.

Ситуационный подход

Ситуационный подход находит широкое применение в сфере организационного управления, автоматизации сложных технических систем, а также в системах искусственного интеллекта. Его истоки правомерно связывать с работой Дж. Маккарти [37], в которой развивается *ситуационное исчисление* — формальный аппарат для построения модели мира и выполнения рассуждений о воздействиях на мир.

Ситуационный подход также актуален для проблематики агентно-ориентированного проектирования систем [27].

Многие исследователи отождествляют понятия ситуации и состояния системы (см., например, работы [38, 39]). Так, у Д. А. Поспелова [39] находим следующую формулировку: "текущая ситуация — совокупность всех сведений о структуре объекта и его функционировании в данный момент времени". Однако такое определение ситуации малопродуктивно для МАС.

Как правило, агент не располагает полной информацией об окружающей среде (ОС). В связи с этим Д. А. Поспелов в работе [40] специально вводит понятие *открытых сред*, предполагающее отказ от постулата полноты знаний у агента и введение локальных описаний среды обитания. Согласно работе [41] к числу фундаментальных свойств ИА относятся способность восприятия среды с *ограниченным разрешением* и способность строить *локальное* представление этой среды на основе ее восприятия. Именно такой случай имеет место в задачах прохождения агентами неизвестного лабиринта.

Агенты воспринимают лишь ограниченную часть ОС и руководствуются лишь локальным представлением о ней, которое выражается понятием ситуации. Ситуация характеризуется совокупностью некоторых динамических факторов (условий и обстоятельств), которые воспринимаются агентом в данное конкретное время и определяют его дальнейшее поведение.

Примеры. Следующие примеры демонстрируют возможные дескрипторы ситуаций:

1. Исходная ситуация: агент вошел в неразведанный коридор и начинает перемещаться вдоль него. Возможны два события в процессе его следования по коридору: обнаружил вход в помещение, либо оказался в тупике. Каждое из этих событий порождает новую ситуацию, в которой необходимо принимать определенное решение по дальнейшим действиям.

2. Для агентов, осуществляющих групповой поиск выхода из лабиринта, текущая ситуация определяется местонахождением всех членов МАС и картой, составленной для разведанной части лабиринта.

3. Для агента в мире Вампуса текущая ситуация определяется текущими показаниями датчиков и данными, накопленными агентом в процессе блуждания по пещере.

Простейшая ситуационная модель поведения агентов является статической и строится на основе продукций "ситуация—действие". Однако для планирования поведения ИА необходима динамическая ситуационная модель, дополненная событийным механизмом контроля ситуаций. В этом случае цикл управления поведением агента, инициированный некоторым событием, включает следующие этапы: анализ ситуации — генерация локальной цели — планирование действий — исполнение плана.

Фактические результаты действий агента могут отличаться от запланированных вследствие наступления неблагоприятных событий. В этом случае осуществляется корректировка локальной цели и соответствующего локального плана действий.

Ситуационные модели поведения агентов

Обсудим методологические аспекты ситуационного моделирования поведения ИА.

Будем полагать, что МАС — группа из n равноправных агентов.

Введем обозначения:

- X, A, U — соответственно пространство состояний, алфавит (конечное множество) действий и множество стратегий поведения каждого агента в МАС;
- X_T, A_T, U_T — соответственно пространство состояний, алфавит действий и множество стратегий поведения группы агентов.

Пусть A^* — множество всех строк в алфавите A . Возможные стратегии $u \in U$ будем рассматривать в качестве строк в алфавите A , а U — как формальный язык в данном алфавите:

$$U \subseteq A^*.$$

Аналогично определяется множество групповых стратегий U_T :

$$U_T = (A_T)^*.$$

Если агенты действуют автономно, то $X_T = (X)^n$ и $A_T = A^n$.

Поведение МАС подчиняется поставленным локальным целям и зависит от процессов в окружающей среде.

Введем дополнительные обозначения:

- X_W, Z — соответственно пространство состояний ОС и ситуационное пространство МАС;
- E_W — множество возможных событий во внешнем мире.

Взаимодействие агентов с ОС описывается уравнениями вида

$$x_W(t+1) = f_W(x_W(t), e(t), a_T(t)); \quad (5)$$

$$x_T(t+1) = f_T(x_W(t), x_T(t), a_T(t)), \quad (6)$$

где $x_W \in X_W$ — состояние ОС; $x_T \in X_T$ — состояние системы агентов; $e \in E_W$ — событие во внешней среде; $a_T \in A_T$ — кортеж групповых действий агентов; $f_W: X_W \times E_W \times A_T \rightarrow X_W$, $f_T: X_W \times X_T \times A_T \rightarrow X_T$ — функции перехода состояний ОС и МАС.

Процессы в системе (5)—(6) протекают в абстрактном времени

$$t \in \{0, 1, 2, \dots\}.$$

По характеру отсчета дискретного времени процессы могут быть синхронными и асинхронными. В первом случае моменты времени определяются принудительно синхронизирующими сигналами. Во втором случае ход времени связан с изменениями в мире.

Текущая ситуация $z(t) \in Z$ определяется состоянием ОС и МАС:

$$z(t) = g(x_W(t), x_T(t)),$$

где $g: X_W \times X_T \rightarrow Z$ — оператор идентификации ситуаций.

Далее Y — множество локальных целей (пространство целей), а X_G — пространство состояний главной задачи, под которым будем понимать булеан множества Y , т.е. множество всех его подмножеств. Тогда $x_G(t) \in X_G$ — множество всех локальных целей, достигнутых к моменту времени t .

Необходимость в изменении локальной цели возникает при наступлении определенных ситуаций. Положим, такая ситуация возникла в момент времени $t = t'$. Процедура целеполагания описывается моделью вида

$$y(t') = f_G(x_G(t'), z(t')),$$

где x_G — состояние задачи, $z \in Z$ — ситуация, $y \in Y$ — новая локальная цель; $f_G: X_G \times Z \rightarrow Y$.

Выбор стратегий поведения ИА зависит от новой цели и текущей ситуации:

$$u_{T|t'+1} = h(y(t'), z(t')), \quad h: Y \times Z \rightarrow U_T.$$

Функционирование МАС определяется следующими факторами:

- 1) начальной ситуацией $z(0)$;
- 2) событиями в ОС $e(0), e(1), e(2), \dots$;
- 3) действиями агентов $a_T(0), a_T(1), a_T(2), \dots$.

Эффективным средством описания ситуаций является логика предикатов первого порядка.

Ситуационную модель системы можно представить *ситуационным графом*. Для этого необходимо расширить число типов контролируемых событий. Пусть E_G и E_T — множества событий, связанных с процессами целеполагания и действиями агентов соответственно. Введем также множество всех возможных событий без дифференциации их типа:

$$E = E_G \cup E_W \cup E_T.$$

В ситуационном графе вершины представляют ситуации, а ориентированные ребра означают события $e \in E$ и отражают смену ситуаций.

Координация действий агентов

Каждый агент имеет доступ к совместно формируемой карте лабиринта, на которой отображается их текущее местоположение. Данная карта является динамической компонентой его базы знаний. Агенты посредством доски объявлений обмениваются информацией о своих намерениях, чтобы избежать конфликтных ситуаций, например, встречного движения по узкому коридору.

Процедура анализа лабиринта направлена на поэтапное составление его карты. На ее основе осуществляется планирование маршрутов перемещений агентов. Эффективность совместных действий агентов оценивается временем прохождения лабиринта и общей длиной пройденных маршрутов. Процесс координации действий агентов в лабиринте должен быть направлен на согласование перемещений агентов с целью минимизировать эти показатели.

Задачи оптимальной маршрутизации

Многие задачи для лабиринтов с известной конфигурацией сводятся к базисной задаче построения кратчайшего маршрута, соединяющего заданные точки лабиринта. Для ее решения могут быть использованы известные результаты в области планирования траекторий движения мобильных роботов, которые достаточно полно освещены в работах [42—45]. Это касается и проблематики совместного планирования прохождения лабиринта группой агентов — здесь также следует учитывать результаты исследования вопросов планирования и координации движения для группы МР (см., например, [46—48]).

Среди алгоритмов построения кратчайшего маршрута в лабиринте [49] весьма популярным является волновой алгоритм (алгоритм Ли) [50, 51]. Однако данный алгоритм оказывается трудоемким для лабиринтов с большим числом ячеек и поэтому малоприменим для задач, в которых вследствие динамически изменяющихся условий требуется оперативно прокладывать новые маршруты. В этом случае предлагается исходную оптимизационную задачу для клетчатого лабиринта редуцировать к более простой задаче поиска кратчайшего маршрута на взвешенных графах. Идея такого сведения заключается в следующем.

Полагаем, что лабиринт состоит из помещений — залов и соединяющих их коридоров. В залах могут находиться препятствия, которые необходимо обходить. Положим, что в каждом проходном помещении построены кратчайшие внутренние трассы, соединяющие различные входы помещения. Данная задача может решаться с помощью волнового алгоритма или других известных алгоритмов, например, *метода графа видимости* (visibility graphs) или *метода декомпозиции на ячейки* (cell decom-

position) [42—45]. Здесь можно отметить также алгоритмы, обсуждаемые в работе [52].

Построенные трассы будем рассматривать как сегменты геометрического графа: его вершины соответствуют точкам входа в помещения, а ребра — построенным трассам. Данный граф, по сути, является *дорожной картой* (roadmap) [43]. Далее решается задача поиска кратчайших маршрутов по дорожной карте каким-либо стандартным алгоритмом (см., например, [49, 53]).

Задача оптимальной маршрутизации возникает также на этапе распознавания лабиринта — при прохождении агентами через разведанную часть лабиринта к неразведанным помещениям.

Выводы

Задача анализа и прохождения неизвестной среды блуждающими по нему автоматами является предметом изучения теоретической информатики. Весьма популярной геометрической моделью такой среды является лабиринт.

Многие принципиальные аспекты алгоритмизации поведения МР могут быть изучены посредством компьютерного моделирования лабиринтного поиска. В статье обсуждается концепция виртуальных лабиринтов и обитающих в них виртуальных роботов.

Дан краткий обзор результатов исследований поведения конечных автоматов в лабиринтах.

Проведен анализ базовых функции и архитектуры ИА, способных решать широкий класс лабиринтных задач.

Исследован вопрос построения ситуационных моделей поведения ИА. Простейшая ситуационная модель поведения агентов является статической и строится на основе продукций "ситуация—действие". Однако для планирования поведения ИА необходима динамическая ситуационная модель, дополненная событийным механизмом контроля ситуаций. В этом случае цикл управления поведением агента включает следующие этапы: анализ ситуации — генерация локальной цели — планирование действий — исполнение плана. Предложена формальная схема построения математических моделей процессов ситуационного управления и планирования действий интеллектуальных агентов.

Приводятся соображения по координации действий агентов. Предложен метод оптимизации маршрутов движения в лабиринте, основанный на сведениях исходной оптимизационной задачи для клетчатого лабиринта к более простой задаче поиска кратчайшего маршрута на взвешенных графах.

Список литературы

1. Кудрявцев В. Б., Килибарда Г., Ушчумлич Ш. М. О поведении автоматов в лабиринтах // Дискретная математика. 1992. Т. 4, Вып. 3. С. 3—28.

2. **Кудрявцев В. Б., Килибарда Г., Ушчумлич Ш. М.** Независимые системы автоматов в лабиринтах // Дискретная математика. 2003. Т. 15, Вып. 2. С. 3—39.
3. **Килибарда Г., Кудрявцев В. Б., Ушчумлич Ш. М.** Коллективы автоматов в лабиринтах // Дискретная математика. 2003. Т. 15, Вып. 3. С. 3—39.
4. **Кудрявцев В. Б., Килибарда Г., Ушчумлич Ш. М.** Системы автоматов в лабиринтах // Интеллектуальные системы. 2006. Т. 10, Вып. 1—4. С. 449—562.
5. **Маргынченко Ю. Г.** Динамика мобильных роботов // Соросовский образовательный журнал. 2000. Т. 6, № 5. С. 110—116.
6. **Micromouse** — конкурс для маленьких роботов. URL: <https://geektimes.ru/post/133577/>.
7. **Рейнер П.** Роботы, лабиринты и архитектура поглощения. URL: <http://pandia.ru/text/79/049/45546.php>.
8. **Shannon C. E.** Presentation of a Maze-Solving Machine // Cybernetics Trans. of the 8th Conf. of the Josiah Macy Jr. Found / Editor: H. Forester, 1951. P. 173—180.
9. **Гарднер М.** Математические головоломки и развлечения. М.: Мир, 1999. 447 с.
10. **Басакер Р., Саати Т.** Конечные графы и сети. М.: Наука, 1974. 368 с.
11. **Freeman H.** On the Encoding of Arbitrary Geometric Configurations // IEEE Trans. Electron. Comput. 1961. Vol. 10, N. 2. P. 260—268.
12. **Дьорр К.** Automaten in Labirinth. I // Elektronische Informationsverarbeitung und Kybernetik. 1971. Vol. 7, N. 2. P. 79—94.
13. **Дьорр К.** Automaten in Labirinth. II // Elektronische Informationsverarbeitung und Kybernetik. 1971. Vol. 7, N. 3. P. 167—190.
14. **Budach L.** Automata and Labyrinths // Mathematische Nachrichten. 1978. Vol. 86. P. 195—282.
15. **Müller H.** Automata Catching Labyrinths with at Most Three Components // Elektronische Informationsverarbeitung und Kybernetik. 1979. Vol. 15, N. 1/2. P. 3—9.
16. **Зыричев А. Н.** О синтезе автомата, обходящего плоские лабиринты с ограниченными дырами // Дискретная математика. 1991. Т. 3, Вып. 1. С. 105—113.
17. **Fischer P. C.** Multi-Tape and Infinite-State Automata: A survey // Communication of the ACM. 1965. Vol. 8, N. 12. P. 799—805.
18. **Antelmann H., Budach L., Rollik H. A.** On Universale Traps // Elektronische Informationsverarbeitung und Kybernetik. 1979. Vol. 15, No. 3. P. 123—131.
19. **Blum M., Kozen D.** On the Power of the Compass // Proc. 19th Annual Symposium on Foundations of Computer Sci. 1978. C. 132—142.
20. **Hoffmann F.** One pebble does not suffice to search plane labyrinths // Lecture Notes of Computer Science. 1981. Vol. 117. P. 433—444.
21. **Ахтеров А. В., Кирильченко А. А., Павловский В. Е., Рогозин К. В.** Способы управления распределенной мобильной системой в условиях неопределенности // Препринты ИПМ им. М. В. Келдыша. 2012. № 67. 32 с. URL: <http://library.keldysh.ru/preprint.asp?id=2012-67>.
22. **Parsons T. D.** Pursuit-Evasion in a Graph / In Theory and Applications of Graphs. Springer-Verlag. 1976. P. 426—441.
23. **Megiddo N., Hakimi S. L., Garey M. R., Johnson D. S., Papadimitriou C. H.** The Complexity of Searching a Graph // Journal of the ACM. 1988. Vol. 35, N. 1. P. 18—44.
24. **Степкин А. В.** Использование коллектива агентов для распознавания графа // Компьютерные исследования и моделирование. 2013. Т. 5. № 4. С. 525—532.
25. **Сапунов С. В.** Восстановление графа с помеченными вершинами перемещающимся по нему мобильным агентом // Известия Саратовского ун-та. Новая серия. Сер. Математика. Механика. Информатика. 2015. Т. 15, Вып. 2. С. 228—238.
26. **Теряев Е. Д., Петрин К. В., Филимонов А. Б., Филимонов Н. Б.** Агентные технологии в автоматизированных информационно-управляющих системах. Ч. I. Основы агентного подхода // Мехатроника, автоматизация, управление. 2010. № 7. С. 11—27.
27. **Теряев Е. Д., Петрин К. В., Филимонов А. Б., Филимонов Н. Б.** Агентные технологии в автоматизированных информационно-управляющих системах. Ч. II. Агентные решения в задачах контроля и управления // Мехатроника, автоматизация, управление. 2010. № 10. С. 25—34.
28. **Макаров И. М., Лохин В. М., Манько С. В., Романов М. П., Крюченко Е. Н., Худак Ю. И., Кучерский Р. В.** Модели и алгоритмы планирования действий и распределения заданий в мультиагентных робототехнических системах // Мехатроника, автоматизация, управление. 2012. № 5. С. 44—50.
29. **Jennings N. R., Wooldridge M. J.** Agent Technology. Berlin; Heidelberg; New-York: Springer-Verlag, 1998.
30. **Wooldridge M. J.** An Introduction to MultiAgent Systems. John Wiley & Sons Ltd, 2002. 366 p.
31. **Рассел С., Норвиг П.** Искусственный интеллект: современный подход. М.: ИД "Вильямс", 2007. 1408 с.
32. **Городецкий В. И., Грушинский М. С., Хабалов А. В.** Многоагентные системы (обзор) // Новости искусственного интеллекта. 1998. № 2. С. 64—116.
33. **Швецов А. Н.** Агентно-ориентированные системы: от формальных моделей к промышленным приложениям // Всероссийский конкурсный отбор обзорно-аналитических статей по приоритетному направлению "Информационно-телекоммуникационные системы". 2008. 101 с.
34. **Ржевский Дж.** Мультиагентные системы в логистике и e-коммерции // International Conference on Intelligent Manufacturing. Wuhan, China, 1995.
35. **Muller J. P.** The Design of Intelligent Agents: A Layered Approach. Springer-Verlag, Berlin. 1996.
36. **Люгер Дж. Ф.** Искусственный интеллект: стратегия и методы решения сложных проблем. М.: Изд. дом "Вильямс", 2005. 864 с.
37. **McCarthy J.** Situations, Actions and Causal Laws // Stanford Artificial Intelligence Project: Memo2, 1963.
38. **Клыков Ю. И.** Ситуационное управление большими системами. М.: Энергия, 1974. 136 с.
39. **Поспелов Д. А.** Ситуационное управление: теория и практика. М.: Наука, 1986. 288 с.
40. **Поспелов Д. А.** От коллектива автоматов к мультиагентным системам // Труды Междунар. семинара "Распределенный искусственный интеллект и многоагентные системы" (DAIMAS'97). СПб., 1997. С. 319—325.
41. **Тарасов В. Б.** Агенты, многоагентные системы, виртуальные сообщества: стратегическое направление в информатике и искусственном интеллекте // Новости искусственного интеллекта. 1998. № 2. С. 5—64.
42. **Sieewart R., Nourbakhsh I. R.** Introduction to Autonomous Mobile Robots. Massachusetts: Bradford Book, 2004. 340 p.
43. **Choset H., Lynch K. M., Hutchinson S., Kantor G., Burgard W., Kavraki L. E., Thrun S.** Principles of Robot Motion: Theory, Algorithms and Implementations. MIT Press, Cambridge, MA, 2005. 603 p.
44. **LaValle S. M.** Planning Algorithms. Cambridge. U. K.: Cambridge University Press, 2006. 1007 p.
45. **Coenen S. A. M., Steinbuch M., van de Molengraff M. J. G., Lunenburg J. J. M., Maus G. J. L.** Motion Planning for Mobile Robots: a Guide. Eindhoven, Eindhoven University of Technology. 2012. 79 p.
46. **Parker L. E.** Path Planning and Motion Coordination in Multiple Mobile Robot Teams. Encyclopedia of Complexity and System Science. R. A. Meyers (ed.). Springer Verlag, 2009. P. 5783—5800.
47. **Desaraju V. R., How J. P.** Decentralized Path Planning for Multi-Agent Teams with Complex Constraints // Autonomous Robots. January 2012. P. 1—19.
48. **Alonso-Mora J.** Collaborative Motion Planning for Multi-Agent Systems, Dr. Diss. Eidgenössische Technische Hochschule ETH Zurich. No 21705. 2014. 176 p.
49. **Кормен Т. Х., Лейзерсон Ч. И., Ривест Р. Л., Штайн К.** Алгоритмы: построение и анализ. М.: Вильямс, 2013. 1328 с.
50. **Rubin F.** The Lee path connection algorithm // IEEE Transactions on Computers. 1974. Vol. C-23, Iss. 9. P. 907—914.
51. **Волновой** алгоритм поиска пути. URL: <http://www.100byte.ru/100btwrks/wv/wv.html>.
52. **Алдошкин Д. Н., Царев Р. Ю.** Поиск пути мобильного робота в условиях наличия препятствий и неполноты информации о среде // Мехатроника, автоматизация, управление. 2016. № 7. С. 465—470.
53. **Изотова Т. Ю.** Обзор алгоритмов поиска кратчайшего пути в графе // Новые информационные технологии в автоматизированных системах. 2016. Вып. 19. С. 341—344.

The Problem of Maze Passage by the Intelligent Agents

A. B. Filimonov, filimon_ab@mail.ru,
Moscow Technological University, Moscow, 119454, Russian Federation,
N. B. Filimonov, nbfilimonov@mail.ru✉,
Lomonosov Moscow State University, Moscow, 119991, Russian Federation
V. Ju. Tikhonov, bot-32@ya.ru,
Moscow Technological University, 119454, Moscow, Russian Federation

Corresponding author: **Filimonov Nikolay B.**, Professor, Dr. Sci. Tech.,
Lomonosov Moscow State University, Moscow, 119991, Russian Federation,
e-mail: nbfilimonov@mail.ru

Received on July, 20, 2016

Asepted on July, 30, 2016

The problems of the analysis and passage of the unknown medium by wandering automata over it is the subject of the study of the theoretical computer science. The maze is very popular geometric model of such medium. The mazes as the artificial testing medium for the experimental research of traffic control algorithms of the autonomous mobile robots are widely used in robotics. The paper is dedicated to the problems of maze passage by robotic intelligent agents (IA). The most principle aspects of the algorithmization of mobile robots conduct may be study by means of computer simulation of maze search. The conception of virtual mazes and virtual robots living in them is discussed in the article. It is given the short review of the results of the theoretical researches of the final automata conduct in chess mazes. They permit to understand the possibilities of mazes passage by the reactive agents. The problem of going around of mazes by the IA occupies the central place in the article. The IA in contrast to the reactive agents are capable to analyze the habitat, to design and optimize its own conduct. In connection with it the statement of the maze problems is changed cardinally. The architecture of the IA which are able to decide the wide class of the maze problems is analyzed. The question of the structure of the situating models of IA conduct occupies the central place in the article. The simplest situating model of the agents' conduct is the static one and it is constructed on the basis of the productions "situation-action". But for the planning of IA conduct it is necessary the dynamic situating model complemented with the event mechanism of the situations control. In this case the cycle of control by the agent conduct has the following stages: the analysis of the situation — the generation of local purpose — the planning of actions — the fulfillment of plan. The formal scheme of the structure of the mathematical models of the situating control processes and the planning of the actions of the IA is proposed. The considerations about the co-ordination of the agents' actions are reduced. The method of the optimization of the routes of the movement in the maze based on the reduction of the initial optimization problems for the checked maze to more simple problem of search of the shortest route of the weigh graphs is suggested.

Keywords: chess maze, automata conduct, intelligent agents, architecture, the situating model of conduct, planning of actions, optimization of routes

Acknowledgements: The work was supported by grant of the Russian Foundation for Basic Research (Project No. 16-08-00313-a).

For citation:

Filimonov A. B., Filimonov N. B., Tikhonov V. Ju. The Problem of Maze Passage by Intelligent Agents, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2016, vol. 17, no. 11, pp. 750—761.

DOI: 10.17587/mau.17.750-761

References

1. **Kudryavcev V. B., Kilibarda G., Ushchumlich Sh. M.** *O povedenii avtomatov v labirintah* (About the behavior of automata in labyrinths), *Diskretnaya Matematika*, 1992, vol. 4, iss. 3, pp. 3—28 (in Russian).
2. **Kudryavcev V. B., Kilibarda G., Ushchumlich Sh. M.** *Nezavisimye sistemy avtomatov v labirintah* (Independent systems of automata in labyrinths), *Diskretnaya Matematika*, 2003, vol. 15, iss. 2, pp. 3—39 (in Russian).
3. **Kilibarda G., Kudryavcev V. B., Ushchumlich Sh. M.** *Kollektivny avtomatov v labirintah* (Collectives of automata in labyrinths), *Diskretnaya Matematika*, 2003, vol. 15, iss. 3, pp. 3—39 (in Russian).
4. **Kudryavcev V. B., Kilibarda G., Ushchumlich Sh. M.** *Sistemy avtomatov v labirintah* (About the behavior of automata in labyrinths), *Intellektual'nye Sistemy*, 2006, vol. 10, iss. 1-4, pp. 449—562 (in Russian).
5. **Martynenko Yu. G.** *Dinamika mobil'nyh robotov* (Dynamics of mobile robots), *Sorosovskij Obrazovatel'nyj Zhurnal*, 2000, vol. 6, no. 5, pp. 110—116 (in Russian).
6. **Micromouse** — konkurs dlya malen'kih robotov (Micromouse — competition for small robots), available at: <https://geektimes.ru/post/133577/>.
7. **Rejner P.** *Roboty, labirinty i arhitektura pogloshcheniya* (Robots, mazes, and subsumption architecture), available at: <http://pandia.ru/text/79/049/45546.php>.
8. **Shannon Cl. E.** Presentation of a Maze-Solving Machine, *Cybernetics Trans. of the 8th Conf. of the Josiah Macy Jr. Found.*, Editor: H. Forester, 1951, P. 173—180.
9. **Gardner M.** *Matematicheskie golovolomki i razvlecheniya* (Mathematical puzzles and entertainment), Moscow, Mir, 1999, 447 p.
10. **Basaker R., Saati T.** *Konechnye grafy i seti* (Finite graphs and networks), Moscow, Nauka, 1974, 368 p.
11. **Freeman H.** On the Encoding of Arbitrary Geometric Configurations, *IEEE Trans. Electron. Comput.*, 1961, vol. 10, no. 2, pp. 260—268.
12. **Döpp K.** Automaten in Labirint. I, *Elektronische Informationsverarbeitung und Kybernetik*, 1971, vol. 7, no. 2, pp. 79—94.
13. **Döpp K.** Automaten in Labirint. II, *Elektronische Informationsverarbeitung und Kybernetik*, 1971, vol. 7, no. 3, pp. 167—190.
14. **Budach L.** Automata and Labyrinths, *Mathematische Nachrichten*, 1978, vol. 86, pp. 195—282.
15. **Müller H.** Automata Catching Labyrinths with at Most Three Components, *Elektronische Informationsverarbeitung und Kybernetik*, 1979, vol. 15, no. 1/2, pp. 3—9.
16. **Zyrichev A. N.** *O sinteze avtomata, obhodyashchego ploskie labirinty s ogranichennymi dyrami* (About the synthesis of the machine, bypassing the flat labyrinths with limited holes), *Diskretnaya Matematika*, 1991, vol. 3, iss. 1, pp. 105—113 (in Russian).
17. **Fischer P. C.** Multi-Tape and Infinite-State Automata: A Survey, *Communication of the ACM*, 1965, vol. 8, no. 12, pp. 799—805.
18. **Antelmann H., Budach L., Rollik H. A.** On Universale Traps, *Elektronische Informationsverarbeitung und Kybernetik*, 1979, vol. 15, no. 3, pp. 123—131.
19. **Blum M., Kozen D.** On the Power of the Compass, *Proc. 19th Annual Symposium on Foundations of Computer Sci.*, 1978, pp. 132—142.

20. **Hoffmann F.** One Pebble does not Suffice to Search Plane Labyrinths, *Lecture Notes of Computer Science*, 1981, vol. 117, pp. 433–444.
21. **Ahterov A. V., Kiril'chenko A. A., Pavlovskij V. E., Rogozin K. V.** *Sposoby upravleniya raspredelennoj mobil'noj sistemoy v usloviyah neopredelennosti* (The control methods by the distributed mobile system in the conditions of uncertainty), *Preprinty IPM im. M. V. Keldysha*, 2012, no. 67, 32 p., available at: <http://library.keldysh.ru/preprint.asp?id=2012-67>.
22. **Parsons T. D.** Pursuit-Evasion in a Graph, *In Theory and Applications of Graphs*, Springer-Verlag, 1976, pp. 426–441.
23. **Megiddo N., Hakimi S. L., Garey M. R., Johnson D. S., Papadimitriou C. H.** The Com-plexity of Searching a Graph, *Journal of the ACM*, 1988, vol. 35, no. 1, pp. 18–44.
24. **Stepkin A. V.** *Ispol'zovanie kollektiva agentov dlya raspoznavaniya grafa* (The use of a team of agents to recognize the graph), *Kompyuternye Issledovaniya i Modelirovanie*. 2013, vol. 5, no. 4, pp. 525–532 (in Russian).
25. **Sapunov S. V.** *Vosstanovlenie grafa s pomechennymi vershinami peremeshhayu-shhimsya po nemu mobil'nym agentom* (Reconstruction of a Labeled Graph by a Graph-walking Mobile Agent), *Izvestiya Saratovskogo un-ta. Novaya seriya. Ser. Matematika. Mekhanika. Informatika*, 2015, vol. 15, iss. 2, pp. 228–238 (in Russian).
26. **Teryaev E. D., Petrin K. V., Filimonov A. B., Filimonov N. B.** *Agentnye tekhnologii v avtomatizirovannykh informacionno-upravlyayushchih sistemah. Ch. I. Osnovy agentnogo podhoda* (Agent Technologies for Automated Information Control Systems. Part I. The Bases of Agent Approach), *Mekhatronika, Avtomatizatsiya, Upravlenie*. 2010, no. 7, pp. 11–27.
27. **Teryaev E. D., Petrin K. V., Filimonov A. B., Filimonov N. B.** *Agentnye tekhnologii v avtomatizirovannykh informacionno-upravlyayushchih sistemah. Ch. II. Agentnye resheniya v zadachah kontrolya i upravleniya* (Agent Technologies for Automated Information-Control Systems. Part II. The Agent Decision in Problems of Control and Management), *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2010, no. 10, pp. 25–34 (in Russian).
28. **Makarov I. M., Lohin V. M., Man'ko S. V., Romanov M. P., Kryuchenkov E. N., Hudak Yu. I., Kucherskiy R. V.** *Modeli i algoritmy planirovaniya dejstvij i raspredeleniya zadaniy v mul'tiagentnykh robototekhnicheskikh sistemah* (Action Planning and Tasks Distribution Models and Algorithms for Multiagent Robotic Systems), *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2012, no. 5, pp. 44–50 (in Russian).
29. **Jennings N. R., Wooldridge M. J.** *Agent Technology*, Berlin; Heidelberg; New-York: Springer-Verlag, 1998.
30. **Wooldridge M. J.** *An Introduction to MultiAgent Systems*, John Wiley & Sons Ltd, 2002, 366 p.
31. **Rassel S., Norvig P.** *Iskusstvennyy intellekt: sovremennyy podhod* (Artificial intelligence: a modern approach), Moscow, Publishing house "Vil'yams", 2007, 1408 p. (in Russian).
32. **Gorodeckij V. I., Grushinskij M. S., Habalov A. V.** *Mnogo-agentnye sistemy (obzor)* (Multi-agent systems (review)), *Novosti Iskusstvennogo Intellekta*, 1998, no. 2, pp. 64–116 (in Russian).
33. **Shvecov A. N.** *Agentno-orientirovannye sistemy: ot formal'nykh modelej k promyshlennym prilozheniyam* (Agent-oriented systems: from formal models to industrial applications), *Vserossijskiy konkursnyy otbor obzorno-analiticheskikh statej po prioritetnomu napravleniyu "Informacionno-telekommunikacionnye sistemy"*, 2008, 101 p. (in Russian).
34. **Rzhevskij Dzh.** *Mul'tiagentnye sistemy v logistike i e-kommercii* (Multi-agent systems in logistics and e-commerce), *International Conference on Intelligent Manufacturing*. Wuhan, China, 1995.
35. **Muller J. P.** *The Design of Intelligent Agents: A Layered Approach*, Springer-Verlag, Berlin, 1996.
36. **Lyuger Dzh. F.** *Iskusstvennyy intellekt: strategiya i metody resheniya slozhnykh problem* (Artificial intelligence: strategy and methods for solving of complex problems), Moscow, Publishing house "Vil'yams", 2005, 864 p. (in Russian).
37. **McCarthy J.** *Situations, Actions and Causal Laws*, Stanford Artificial Intelligence Project: Memo2, 1963.
38. **Klykov Yu. I.** *Situacionnoe upravlenie bol'shimi sistemami* (Situational control of large systems), Moscow, EHnergiya, 1974, 136 p. (in Russian).
39. **Pospelov D. A.** *Situacionnoe upravlenie: teoriya i praktika* (Situational management: theory and practice), Moscow, Nauka, 1986, 288 p.
40. **Pospelov D. A.** *Ot kollektiva avtomatov k mul'tiagentnym sistemam* (From team automata to multi-agent systems), *Trudy Mezh-dunarodnogo seminar "Raspredelennyy iskusstvennyy intellekt i mnogo-agentnye sistemy" (DAIMAS'97)*, SPb., 1997, pp. 319–325 (in Russian).
41. **Tarasov V. B.** *Agenty, mnogoagentnye sistemy, virtual'nye soobshchestva: strategicheskoe napravlenie v informatike i iskusstvennom intellekte* (Agents, multi-agent systems, virtual communities: strategic direction in computer science and artificial intelligence), *Novosti Iskusstvennogo Intellekta*, 1998, no. 2, pp. 5–64 (in Russian).
42. **Siegrwart R., Nourbakhsh I. R.** *Introduction to Autonomous Mobile Robots*. Massachusetts: Bradford Book, 2004, 340 p.
43. **Choset H., Lynch K. M., Hutchinson S., Kantor G., Burgard W., Kavraki L. E., Thrun S.** *Principles of Robot Motion: Theory, Algorithms and Implementations*, MIT Press, Cambridge, MA, 2005, 603 p.
44. **LaValle S. M.** *Planning Algorithms*. Cambridge, U. K.: Cambridge University Press, 2006, 1007 p.
45. **Coenen S. A. M., Steinbuch M., van de Molengraff M. J. G., Lunenburg J. J. M., Maus G. J. L.** *Motion Planning for Mobile Robots: a Guide*. Eindhoven, Eindhoven University of Technology, 2012, 79 p.
46. **Parker L. E.** *Path Planning and Motion Coordination in Multiple Mobile Robot Teams*. Encyclopedia of Complexity and System Science. R. A. Meyers (ed.), Springer Verlag, 2009, pp. 5783–5800.
47. **Desaraju V. R., How J. P.** *Decentralized Path Planning for Multi-Agent Teams with Complex Constraints*, *Autonomous Robots*, January, 2012, pp. 1–19.
48. **Alonso-Mora J.** *Collaborative Motion Planning for Multi-Agent Systems*, Dr. Diss. Eidgenössische Technische Hochschule ETH Zurich, no. 21705, 2014, 176 p.
49. **Kormen T. H., Lejzerson Ch. I., Rivest R. L., Shtajn K.** *Algoritmy: postroenie i analiz* (Algorithms: construction and analysis), Moscow, Vil'yams, 2013, 1328 p. (in Russian).
50. **Rubin F.** *The Lee path connection algorithm*, *IEEE Transactions on Computers*, 1974, vol. C-23, iss. 9, pp. 907–914 (in Russian).
51. **Volnovoj** *algoritim poiska puti* (Ripple pathfinding algorithm), available at: <http://www.100byte.ru/100btwrks/wv/wv.html>.
52. **Aldoshkin D. N., Carev R. Yu.** *Poisk puti mobil'nogo robota v usloviyah nalichiya prepyatstvij i nepolnoty informacii o srede* (Mobile Robot Path Planning in the Presence of Obstacles and Lack of Information about the Environment), *Mekhatronika, avtomatizatsiya, upravlenie*, 2016, no. 7, pp. 465–470 (in Russian).
53. **Izotova T. Yu.** *Obzor algoritmov poiska krachajshogo puti v grafe* (Overview of algorithms for finding the shortest path in the graph), *Novye informacionnye tekhnologii v avtomatizirovannykh sistemah*, 2016, iss. 19, pp. 341–344 (in Russian).

Е. В. Страшнов, вед. программист, strashnov_evg@mail.ru,

М. А. Торгашев, канд. физ.-мат. наук, зав. сектором, mtorg@mail.ru,

Научно-исследовательский институт системных исследований Российской академии наук

Моделирование динамики электроприводов виртуальных роботов в имитационно-тренажерных комплексах¹

Рассматривается задача моделирования динамики электропривода с учетом его паспортных параметров применительно к задачам, решаемым в системах виртуального окружения. Для моделирования системы шарнирно связанных тел в реальном времени с учетом динамики электропривода предлагается использовать метод последовательных импульсов. В данном методе динамика электропривода реализуется с помощью специального ограничения, связывающего координаты тел с углом поворота двигателя. Для реализации сложных механизмов, управляемых одним двигателем, предлагается ввести специальное ограничение "повторителя". Апробация предложенных алгоритмов и методов была проведена в подсистеме динамики имитационно-тренажерного комплекса для моделирования роботов.

Ключевые слова: электропривод, система шарнирно связанных тел, ограничения для электропривода и "повторителя", метод последовательных импульсов, система виртуального окружения

Введение

Для управления виртуальной моделью робота в имитационно-тренажерных комплексах требуется создать функциональную схему, управляющее воздействие в которой формируется через передаточную функцию, содержащую параметры электропривода (пусковой момент двигателя, скорость холостого хода двигателя, коэффициент редукции передаточного механизма и т.д.). Это управляющее воздействие входит в математическую модель динамики робота. При использовании неточной модели, в которой динамика электропривода описывается без учета его параметров, возможно сильное расхождение поведения виртуальной модели робота с его реальным аналогом. Поэтому необходимо реализовать такую математическую модель динамики электропривода с его реальными параметрами, которая позволит отработать основные режимы управления виртуальным роботом в системах виртуального окружения.

Моделирование роботов и манипуляторов сводится к моделированию динамики шарнирно связанных тел, в сочленении которых может быть установлен электропривод. Данный электропривод формирует такое управляющее воздействие, которое позволяет управлять относительным движением двух тел, соединенных шарниром.

Существуют два подхода для моделирования динамики системы связанных тел. Первый основывается на использовании обобщенных координат (относительных углов или перемещений), а второй подход состоит в использовании полного набора координат (абсолютных координат положения и ориентации каждого тела).

Первому подходу посвящено достаточно много публикаций в робототехнике [1–3], где динамика

электропривода учитывается при построении пропорционально-интегрально-дифференцирующего (ПИД) или пропорционально-дифференцирующего (ПД) регулятора для управления параметрами шарнира (координатой и скоростью). В этих публикациях математическая модель электродвигателя содержит параметры (сопротивление обмоток, постоянная крутящего момента), отличные от его паспортных данных (пусковой момент, скорость холостого хода и электромеханическая постоянная времени [4]). Обычно разработчикам виртуальной модели робота и функциональной схемы для его управления недоступны данные параметры. Поэтому одна из задач, решаемая в данной статье, состоит в том, чтобы построить математическую модель электродвигателя с использованием его паспортных данных.

При моделировании системы шарнирно связанных тел с использованием второго подхода затруднительно непосредственно использовать математическую модель динамики электропривода, так как движение звеньев определяется через их абсолютные координаты, а динамика электропривода определяется через относительные координаты. В публикации [5] для решения задачи компьютерной анимации игровых персонажей используется ПД регулятор, который задается в виде дифференциального уравнения относительно угла поворота двух соединенных шарниром тел, которое имеет аналитическое решение. Далее это решение используется в методе моделирования шарнирно связанных тел на основе импульсов и методов инверсной динамики. Данный подход является достаточно трудоемким и требует значимых вычислительных затрат. В диссертациях [6, 7] рассматривается упрощенная модель электродвигателя, динамика которого реализуется с помощью специального ограничения. В данном ограничении задается желаемая скорость относительного движения в шарнире и модуль

¹ Данная работа выполнена при поддержке грантов РФФИ, проекты № 16-07-01104, № 16-37-00107 мол_а.

максимального момента, развиваемого двигателем. Такой подход только имитирует поведение электродвигателя, поэтому пригоден лишь в игровых приложениях. Задача, решаемая в данной статье, состоит в разработке методов моделирования динамики системы шарнирно связанных тел на основе полного набора координат с использованием математической модели электропривода. Данные методы являются новыми в рамках моделирования робототехнических средств в имитационно-тренажерных комплексах.

В этой работе предлагается использовать метод последовательных импульсов [8—10] для моделирования динамики системы связанных тел. Для реализации динамики электропривода выводится специальное ограничение, которое связывает координаты двух тел и угол поворота двигателя. Кроме того, для того чтобы смоделировать сложные механизмы, такие как губки захвата манипулятора, предлагается ввести ограничение "повторителя" с аналогичной структурой. Данное ограничение позволит смоделировать плоскопараллельное движение губок манипулятора на основе управления от одного двигателя. Предлагаемые алгоритмы и методы моделирования системы связанных тел с учетом динамики электропривода реализованы и апробированы в подсистеме динамики имитационно-тренажерного комплекса, разработанного в ФГУ ФНЦ НИИСИ РАН.

Имитационно-тренажерный комплекс

Имитационно-тренажерный комплекс предназначен для моделирования сложных динамических систем и тренировки операторов по выполнению операций с помощью компьютерной симуляции. Эти динамические системы включают колесные и гусеничные роботы, а также другие управляемые механизмы (например, лифты). Виртуальные 3D-модели роботов и окружающей обстановки создаются в системе трехмерного моделирования Autodesk 3ds Max. Подготовленные модели загружаются в систему динамики и систему визуализации. Для управления движением роботов используются реальные или виртуальные пульта. Воздействия оператора на элементы управления (кнопки, джойстики, тумблеры и т.д.) этих пультов передаются в функциональную схему, которая вычисляет управляющие сигналы — напряжения, подаваемые на электродвигатели робота. Далее подсистема динамики рассчитывает скорости и новые координаты динамических объектов, а также скорости и углы поворотов двигателей. Новые значения передаются по специальному информационному протоколу в систему визуализации, которая синтезирует изображение виртуальной сцены с новыми положениями и ориентациями объектов. Один шаг моделирования и визуализации занимает не более 40 мс, что

обеспечивает режим реального времени (не менее 25 кадров в секунду) и позволяет получить непрерывное и гладкое движение виртуальных объектов.

Математическая модель динамики электропривода

Для начала получим математическую модель динамики электродвигателя постоянного тока без редуктора и учета нагрузки. Рассматривается идеализированный двигатель с прямолинейной механической характеристикой, в роли индуктора которого используются постоянные магниты. Из электрической схемы для якоря двигателя можно получить его механическую характеристику [1]:

$$\tau_m = \frac{K_t}{R} U - \frac{K_t^2}{R} \omega_m, \quad (1)$$

где τ_m — крутящий момент двигателя; U — напряжение, приложенное к якорю двигателя; ω_m — угловая скорость вращения якоря двигателя; K_t — постоянная крутящего момента (определяется силой магнитного поля, числом обмоток якоря и т.д.); R — сопротивление обмоток якоря двигателя.

Напряжение U ограничено по модулю величиной U_m , т.е. $|U| \leq U_m$, где $U_m > 0$. Рассмотрим теперь вместо напряжения U его нормированную величину $u = \frac{U}{U_m}$, где $|u| \leq 1$. Тогда выражение (1) переписывается в следующем виде:

$$\tau_m = \frac{K_t U_m}{R} \left(u - \frac{K_t}{U_m} \omega_m \right). \quad (2)$$

Перейдем от параметров, используемых в формуле (2), к паспортным данным электродвигателя [4]: пусковому моменту M_{Π} и скорости холостого хода $\omega_{\text{ХХ}}$. Пуск двигателя осуществляется при $u = \pm 1$ и $\omega_m = 0$, что соответствует значению момента $\tau_m = \pm M_{\Pi}$. В установившемся режиме, когда $\tau_m = 0$ и $u = \pm 1$ двигатель разгоняется до скорости $\omega_m = \pm \omega_{\text{ХХ}}$. Делая в выражении (2) замены $M_{\Pi} = \frac{K_t U_m}{R}$ и $\omega_{\text{ХХ}} = \frac{U_m}{K_t}$, получим механическую характеристику двигателя, выраженную через его паспортные данные:

$$\tau_m = M_{\Pi} \left(u - \frac{\omega_m}{\omega_{\text{ХХ}}} \right). \quad (3)$$

Тогда основное уравнение динамики электродвигателя примет вид

$$I_m \dot{\omega}_m = \tau_m = M_{\Pi} \left(u - \frac{\omega_m}{\omega_{\text{ХХ}}} \right), \quad (4)$$

где I_m — момент инерции якоря электродвигателя.

Уравнение (4) дополняется кинематическим соотношением, связывающим угол поворота двигателя α_m с его угловой скоростью:

$$\dot{\alpha}_m = \omega_m. \quad (5)$$

Если момент инерции I_m неизвестен, то его можно вычислить через электромеханическую постоянную времени t_m , которая определяет время разгона двигателя до скорости холостого хода ω_{xx} при постоянном крутящем моменте $\tau_m = M_{\Pi}$ и постоянном значении напряжения $u = 1$. Интегрируя выражение (4) по t от 0 до t_m при $u = 1$ с $\tau_m = M_{\Pi}$, получим $\frac{I_m \omega_{xx}}{t_m} = M_{\Pi}$, откуда $I_m = \frac{M_{\Pi} t_m}{\omega_{xx}}$.

В действительности время разгона двигателя больше, чем t_m , так как крутящий момент τ_m вида (3) уменьшается при увеличении скорости двигателя. Поэтому интегрируя формулу (4) со значением напряжения $u = 1$, получим [4]

$$\omega_m(t) = \omega_{xx} \left(1 - e^{-\frac{t}{t_m}} \right),$$

откуда следует, что $\omega_m(t) \rightarrow \omega_{xx}$ при $t \rightarrow \infty$.

Теперь перейдем к описанию динамики электропривода. Для этого рассмотрим упрощенную модель, когда двигатель управляет только одним звеном (нагрузкой) без учета динамики других звеньев (рис. 1).

Если электродвигатель создает малый крутящий момент, но вращается с высокой угловой скоростью, то между ним и звеном добавляется редуктор (шестерочный механизм передачи движения от электродвигателя к звену) с коэффициентом редукции $r \geq 1$ (в отсутствие редуктора $r = 1$).

При использовании механизма передачи с двумя шестеренками $r = \frac{n_2}{n_1}$, где n_1 — число зубьев входной шестеренки, n_2 — число зубьев выходной шестеренки. Тогда угловая скорость и крутящий момент нагрузки будут определяться через угловую ско-

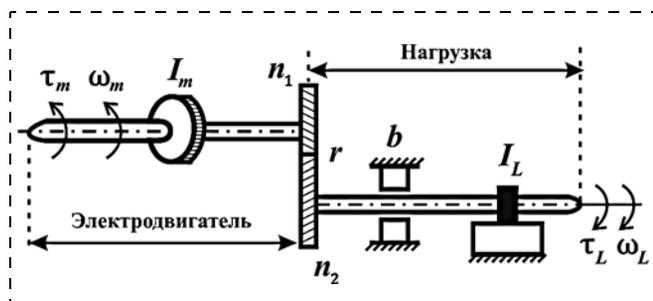


Рис. 1. Структура электропривода с нагрузкой

рость и крутящий момент электродвигателя следующим образом:

$$\omega_L = \frac{1}{r} \omega_m; \tau_L = r \tau_m, \quad (6)$$

где ω_L — угловая скорость нагрузки; τ_L — момент, действующий на звено.

Выведем уравнение динамики для нагрузки с учетом динамики электродвигателя. Добавляя в уравнение (4) момент от нагрузки, передаваемый через редуктор, согласно второму уравнению в (6) получим

$$I_m \dot{\omega}_m = M_{\Pi} \left(u - \frac{\omega_m}{\omega_{xx}} \right) - \frac{1}{r} \tau_L. \quad (7)$$

Динамика нагрузки определяется моментом, приложенным к звену, а также трением в подшипниках с коэффициентом трения b . Поэтому уравнение динамики нагрузки примет вид

$$I_L \dot{\omega}_L = \tau_L - b \omega_L,$$

где I_L — приведенный момент инерции нагрузки относительно ее оси вращения.

Подставляя первое уравнение (6) в выражение (7), получим математическую модель динамики электропривода, управляющего одним звеном:

$$(I_L + r^2 I_m) \dot{\omega}_L = r M_{\Pi} \left(u - \frac{r \omega_L}{\omega_{xx}} \right) - b \omega_L. \quad (8)$$

Уравнение вида (8) имеет схожую структуру с математическими моделями, полученными в работах [1, 2], которые позволяют с требуемой точностью моделировать динамику виртуальных роботов и манипуляторов в имитационно-тренажерных комплексах при использовании обобщенных координат.

Динамика шарнирно связанных тел

Моделирование динамики роботов осуществляется на основе их представления в виде шарнирно связанных тел. В статье [8] разработан метод последовательных импульсов с описанием системы тел с помощью полного набора координат (абсолютного положения и ориентации каждого тела). Данный метод заключается в последовательном вычислении таких импульсов, которые стремятся обеспечить выполнение ограничений, накладываемых на скорости тел. В работах [8, 9] разработан метод отдельных импульсов, который осуществляет стабилизацию ограничений относительно координат тел. Метод последовательных импульсов со стабилизацией ограничений на основе отдельных импульсов является итеративным, а для обеспечения реального времени используются специальные критерии окончания итераций [10].

В статьях [8–10] достаточно подробно описано, как составлять различные типы ограничений: в виде равенств и неравенств, а также ограничения для учета трения. Выведем теперь ограничение для

электропривода и специальное ограничение "повторителя".

На рис. 2 изображено осевое шарнирное соединение двух тел с номерами i и j , относительное движение в котором управляется с помощью электропривода. Для учета динамики электропривода рассматриваются такие ограничения, которые накладываются не только на координаты тел, но и на угол поворота двигателя α_m , который вычисляется с помощью уравнения (5). Данные ограничения будут иметь следующий вид:

$$G(\mathbf{X}_i, \mathbf{X}_j, \alpha_m) = 0, \quad (9)$$

где \mathbf{X}_i и \mathbf{X}_j — координаты, задающие положения и ориентации i -го и j -го тел (например, координаты центра масс и кватернион).

В соотношении (9) входят такие ограничения, в которых содержатся только координаты, задающие ориентацию тел. Поэтому после дифференцирования (9) получим ограничение в терминах угловых скоростей тел (без линейных скоростей) и угловой скорости двигателя:

$$\dot{G} = \mathbf{J}_i \boldsymbol{\omega}_i + \mathbf{J}_j \boldsymbol{\omega}_j + J_m \omega_m = 0, \quad (10)$$

где \mathbf{J}_i и \mathbf{J}_j — строки-якобианы ограничения для тел размерности 1 на 3; J_m — якобиан ограничения для двигателя; $\boldsymbol{\omega}_i$ и $\boldsymbol{\omega}_j$ — угловые скорости тел.

Сформируем ограничения (9) и (10) для учета динамики электропривода. Задача состоит в том, чтобы обеспечить следующее соотношение между относительным углом в шарнире α (рис. 2), отсчитываемым против часовой стрелки, и углом поворота двигателя:

$$\alpha - \frac{1}{r} \alpha_m = 0.$$

Запишем это ограничение в виде

$$\begin{aligned} G_m &= \sin\left(\frac{\alpha_m}{r} - \alpha\right) = \\ &= \sin\left(\frac{\alpha_m}{r}\right) \cos\alpha - \cos\left(\frac{\alpha_m}{r}\right) \sin\alpha = 0. \end{aligned} \quad (11)$$

Для описания шарнира введем две системы координат (рис. 2), имеющих общее начало в точке P : одна из них жестко связана с i -м звеном и определяется ортами $\{\mathbf{n}_{i0}, \mathbf{n}_{i1}, \mathbf{n}_{i2}\}$, а другая жестко связана с j -м звеном и определяется ортами $\{\mathbf{n}_{j0}, \mathbf{n}_{j1}, \mathbf{n}_{j2}\}$ (для наглядности введенные системы координат изображены на рис. 2 вне точки шарнира). При моделировании эти системы координат в начальный момент времени совпадают, а орты вычисляются через абсолютные координаты двух тел \mathbf{X}_i и \mathbf{X}_j соответственно. Осевой шарнир определяется таким образом, что орты \mathbf{n}_{i0} и \mathbf{n}_{j0} совпадают и соответствуют оси вращения. Поэтому, используя геометри-

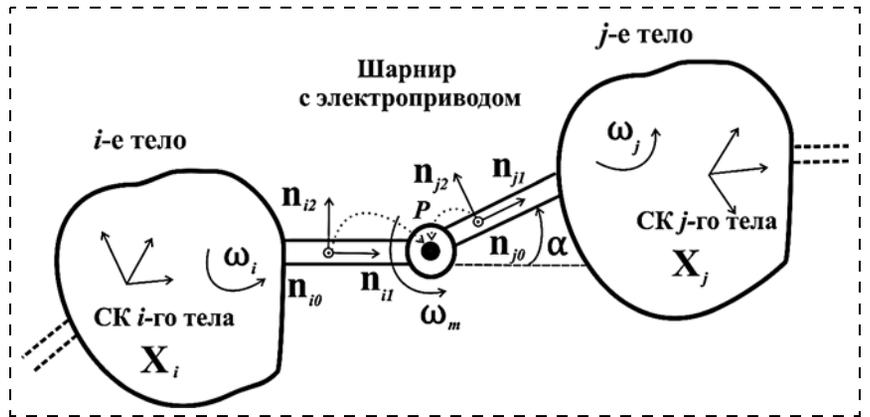


Рис. 2. Шарнирное соединение двух тел с электроприводом

ческое определение скалярного произведения, косинус и синус угла α можно вычислить с помощью данных ортов следующим образом:

$$\cos\alpha = \mathbf{n}_{i1} \mathbf{n}_{j1}; \quad \sin\alpha = -\cos\left(\frac{\pi}{2} + \alpha\right) = -\mathbf{n}_{i1} \mathbf{n}_{j2}. \quad (12)$$

Подставляя данные соотношения в формулу (11), получим ограничение вида (9) для моделирования динамики электропривода:

$$G_m = \sin\left(\frac{\alpha_m}{r}\right) \mathbf{n}_{i1} \mathbf{n}_{j1} + \cos\left(\frac{\alpha_m}{r}\right) \mathbf{n}_{i1} \mathbf{n}_{j2} = 0. \quad (13)$$

Дифференцируя (13), получим ограничение вида (10) в терминах скоростей:

$$\dot{G}_m = (\boldsymbol{\omega}_j - \boldsymbol{\omega}_i) \mathbf{w} + \omega_m w_m = 0, \quad (14)$$

где $\mathbf{J}_i = -\mathbf{w}^T$, $\mathbf{J}_j = \mathbf{w}^T$, $J_m = w_m$,

$$\mathbf{w} = (\mathbf{n}_{j1} \times \mathbf{n}_{i1}) \sin\left(\frac{\alpha_m}{r}\right) + (\mathbf{n}_{j2} \times \mathbf{n}_{i1}) \cos\left(\frac{\alpha_m}{r}\right),$$

$$w_m = \frac{1}{r} \left(\mathbf{n}_{i1} \mathbf{n}_{j1} \cos\left(\frac{\alpha_m}{r}\right) - \mathbf{n}_{i1} \mathbf{n}_{j2} \sin\left(\frac{\alpha_m}{r}\right) \right).$$

При дифференцировании были использованы формула Бура [11] $\dot{\mathbf{n}}_* = \boldsymbol{\omega}_* \times \mathbf{n}_*$ (для дифференцирования в мировой системе координат, здесь звездочка означает произвольный индекс), свойство смешанного произведения $\mathbf{a}(\mathbf{b} \times \mathbf{c}) = \mathbf{b}(\mathbf{c} \times \mathbf{a}) = \mathbf{c}(\mathbf{a} \times \mathbf{b})$ и уравнение (5).

Из определения векторного произведения получим, что $\mathbf{n}_{j1} \times \mathbf{n}_{i1} = -\sin\alpha \mathbf{n}_{i0}$ и $\mathbf{n}_{j2} \times \mathbf{n}_{i1} = -\sin(\pi/2 + \alpha) \mathbf{n}_{i0} = -\cos\alpha \mathbf{n}_{i0}$ (рис. 2). Далее, подставляя формулы (12) в выражение для w_m , запишем ограничение (14) в виде

$$\begin{aligned} \dot{G}_m &= -\cos\left(\alpha - \frac{\alpha_m}{r}\right) (\boldsymbol{\omega}_j - \boldsymbol{\omega}_i) \mathbf{n}_{i0} + \\ &+ \omega_m \frac{1}{r} \cos\left(\alpha - \frac{\alpha_m}{r}\right) = 0. \end{aligned}$$

Углы α и α_m находятся в пределах $-\pi \leq \alpha \leq \pi$ и $-r\pi \leq \alpha_m \leq r\pi$, а в начальный момент времени

$\alpha = \alpha_m = 0$. Если ограничение (11) будет выполнено, то $\sin\left(\frac{\alpha_m}{r} - \alpha\right) = 0$. Поэтому, поделив данное выражение на $\cos\left(\alpha - \frac{\alpha_m}{r}\right) \neq 0$, получим эквивалентное ограничение

$$\dot{G}_m = -(\omega_j - \omega_i)\mathbf{n}_{i0} + \omega_m \frac{1}{r} = 0.$$

Данное соотношение совпадает с первым уравнением в (6), так как скорость ω_L определяет относительную скорость между телами, т. е. $\omega_L = (\omega_j - \omega_i)\mathbf{n}_{i0}$.

Теперь рассмотрим специальный тип ограничения "повторителя". В некоторых механизмах, таких как двухстепенная механическая рука (рис. 3, см. третью сторону обложки), предназначенная для захвата предмета, электропривод может управлять одновременно несколькими звеньями. Звено, которым управляет электропривод, называется основным, остальные управляемые звенья называются "повторителями". Для описания движения "повторителя" вводится специальное ограничение, связывающее его угол поворота с углом поворота двигателя. Такая связь имеет вид

$$\alpha - \frac{k}{r}\alpha_m = 0,$$

где $k \neq 0$ — коэффициент "повторителя" (для губок руки захвата $k = -1$).

Аналогично выражению (13), используя соотношение (12), ограничение "повторителя" можно записать в виде

$$\begin{aligned} G_r &= \sin\left(\frac{k\alpha_m}{r} - \alpha\right) = \\ &= \sin\left(\frac{k\alpha_m}{r}\right)\mathbf{n}_{i1}\mathbf{n}_{j1} + \cos\left(\frac{k\alpha_m}{r}\right)\mathbf{n}_{i1}\mathbf{n}_{j2} = 0. \end{aligned} \quad (15)$$

Относительно скоростей тел и скорости двигателя ограничение вида (10) после дифференцирования (15) примет вид

$$\dot{G}_r = (\omega_j - \omega_i)\mathbf{w}' + \omega_m \mathbf{w}'_m = 0, \quad (16)$$

где $\mathbf{J}_i = -\mathbf{w}'^T$, $\mathbf{J}_j = \mathbf{w}'^T$, $\mathbf{J}_m = \mathbf{w}'_m$,

$$\mathbf{w}' = (\mathbf{n}_{j1} \times \mathbf{n}_{i1})\sin\left(\frac{k\alpha_m}{r}\right) + \mathbf{n}_{j2} \times \mathbf{n}_{i1}\cos\left(\frac{k\alpha_m}{r}\right),$$

$$\mathbf{w}'_m = \frac{k}{r}\left(\mathbf{n}_{i1}\mathbf{n}_{j1}\cos\left(\frac{k\alpha_m}{r}\right) - \mathbf{n}_{i1}\mathbf{n}_{j2}\sin\left(\frac{k\alpha_m}{r}\right)\right).$$

Выполняя с формулой (16) преобразования, аналогичные сделанным с ограничением (14), получим $\omega_L = \frac{k\omega_m}{r}$.

При численном моделировании динамики шарнирно связанных тел и электропривода используется полунявная схема Эйлера, в которой по явной схеме сначала вычисляются скорости тел и скорость двигателя (на основе (4)) под действием всех известных внешних сил. После вычисления новых скоростей ограничения типа (10) в общем случае нарушаются. Поэтому в методе последовательных импульсов требуется вычислить такой импульс p реакции связи вида (9), который обеспечит выполнение (10) уже относительно новых скоростей. Согласно [13] и уравнению (7) скорости тел и двигателя изменяются под действием импульса p следующим образом:

$$\omega'_i = \omega_i + p\mathbf{I}_i^{-1}\mathbf{J}_i^T;$$

$$\omega'_j = \omega_j + p\mathbf{I}_j^{-1}\mathbf{J}_j^T;$$

$$\omega'_m = \omega_m + p\mathbf{I}_m^{-1}\mathbf{J}_m.$$

где \mathbf{I}_i^{-1} и \mathbf{I}_j^{-1} — обратные тензоры инерции двух тел.

Подставляя данные соотношения в формулу (10), получим следующее выражение для вычисления импульса p :

$$\begin{aligned} p(\mathbf{J}_i\mathbf{I}_i^{-1}\mathbf{J}_i^T + \mathbf{J}_j\mathbf{I}_j^{-1}\mathbf{J}_j^T + \mathbf{I}_m^{-1}\mathbf{J}_m^2) = \\ = -(\mathbf{J}_i\omega_i + \mathbf{J}_j\omega_j + \mathbf{J}_m\omega_m). \end{aligned}$$

На каждой итерации метода последовательных импульсов скорости тел ω_i , ω_j и скорость двигателя ω_m являются текущими по отношению к данной итерации, поэтому после вычисления скоростей ω'_i , ω'_j и ω'_m ограничение (10) автоматически выполнено. Однако, так как требуется обеспечить выполнение других ограничений, то ограничение (10) будет выполнено с заданной точностью только после проведения нескольких итераций.

Выполнимость ограничений вида (9) обеспечивается с помощью технологии стабилизации на основе отдельных импульсов, с которой можно ознакомиться в работах [8, 9]. После вычисления скоростей тел и двигателя вычисляются новые координаты тел по неявной схеме относительно скоростей.

Результаты моделирования

В качестве апробации разработанных методов моделирования системы шарнирно связанных тел с учетом динамики электропривода рассмотрим мотор-редуктор WG-7152 [14] и проверим на нем адекватность разработанного метода последовательных импульсов, используя ограничения (13) и (14) в задаче управления одним звеном. Выберем режим работы электропривода со следующими параметрами: $r = 50$, $b = 0$, $M_{\Pi} = 0,2$ Н·м, $\omega_{xx} = 49$ мин⁻¹. Приведенный момент инерции нагрузки равен $I_L = 30,833$ кг·м². Максимальная угловая скорость звена равна скорости холостого хода, поделенной на

коэффициент редукции, т.е. $\omega_L^{\max} = 0,1026$ рад/с. Моделирование проводится на основе метода последовательных импульсов с использованием ограничений (13) и (14), с постоянным шагом $\Delta t = 0,01$ с и с постоянным значением напряжения $u = 1$.

На рис. 4 приведена зависимость угловой скорости звена, управляемого двигателем, от времени для двух значений постоянной времени: $t_m = 1$ с (график 1), $t_m = 0,5$ с (график 2). Полученные результаты совпадают с аналитическим решением уравнения (8) с точностью $\varepsilon = 10^{-6}$. Таким образом, разработанные в статье методы моделирования шарнирно связанных тел с учетом динамики электропривода обладают точностью, необходимой для моделирования роботов в имитационно-тренажерных комплексах.

Разработанные методы реализованы в виде программных модулей на языке С++ и внедрены в подсистему динамики имитационно-тренажерного комплекса, созданного в ФГУ ФНЦ НИИСИ РАН. Комплексная апробация описанных методов и ал-

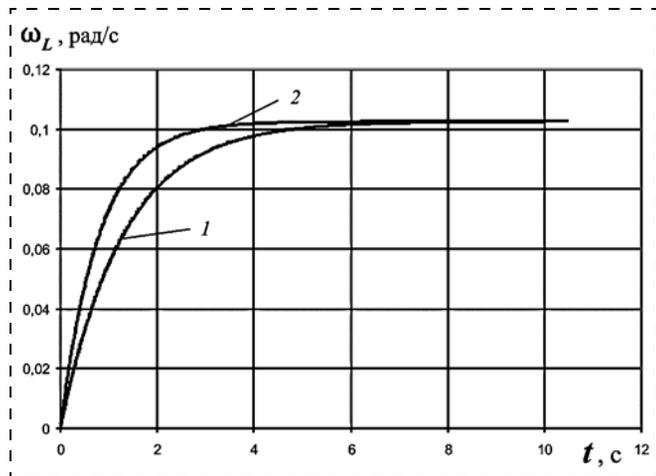


Рис. 4. Зависимость угловой скорости звена ω_L от времени t



Рис. 5. Захват контейнера с помощью робота MF-4

горитмов на многих моделях показала полную их адекватность требованиям систем виртуального окружения. На рис. 5 показана модель робота MF-4, для которого решается задача захвата контейнера с помощью губок манипулятора. Для моделирования динамики электроприводов используются ограничения (13) и (14), а повторяемое движение губки реализуется через ограничения (15) и (16).

Заключение

Предложенные в статье методы и алгоритмы позволяют в масштабе реального времени моделировать динамику электропривода в системах виртуального окружения. Использование паспортных параметров электропривода упрощает создание моделей роботов и построение системы управления на основе функциональной схемы. Апробация разработанных методов в имитационно-тренажерном комплексе показала их применимость для решения задач, связанных с реализацией различных режимов управления роботами. В дальнейшем предложенные алгоритмы и методы могут быть использованы в имитационных комплексах, виртуальных лабораториях, системах моделирования динамики сложных объектов и т.д.

Список литературы

1. Asada H. Introduction to Robotics. Lecture Notes, 2005.
2. Spong M. W., Hutchinson S., Vidyasagar M. Robot Modeling and Control. New York: John Wiley & Sons, Inc., 2006.
3. Тягунов О. А. Математические модели и алгоритмы управления промышленных транспортных роботов // Информационно-измерительные и управляющие системы. 2007. Т. 5, № 5. С. 63—69.
4. Кардонов Г. А. Электрические машины: Курс лекций. СПб.: Санкт-Петербургский государственный институт точной механики и оптики, 2002. 145 с.
5. Weinstein R., Guendelman E., Fedkiw R. Impulse-Based Control of Joints and Muscles. IEEE Transactions on Visualization and Computer Graphics // Ref. Libr. 2008. Vol. 14, N. 1. P. 37—46.
6. Kenny Erleben. Stable, robust, and versatile multibody dynamics animation, PhD thesis, University of Copenhagen, 2004.
7. Steffen Wittmeier. Physics-Based Modeling and Simulation of Musculoskeletal Robots, PhD thesis, Technical University of Munich, 2014.
8. Михайлюк М. В., Страшнов Е. В. Моделирование системы связанных тел методом последовательных импульсов // Труды НИИСИ РАН. 2014. Т. 4, № 2. С. 52—60.
9. Страшнов Е. В., Михайлюк М. В. Моделирование ограничений на относительное движение шарнирно связанных тел // Мехатроника, автоматизация, управление. 2015. Т. 16, № 10. С. 678—685.
10. Михайлюк М. В., Страшнов Е. В. Моделирование динамики системы связанных тел с учетом трения в шарнирах // Наука и Образование. МГТУ им. Н. Э. Баумана. Электрон. журн. 2016. № 1. С. 108—124.
11. Курс теоретической механики: Учебник для вузов / В. И. Дронг, В. В. Дубинин, М. М. Ильин и др.; Под общ. ред. К. С. Колесникова / М.: Изд-во МГТУ им. Н. Э. Баумана, 2005. 736 с.
12. DAGU MK II claw with servo motor. URL: <http://www.electronshtk.ru/item/mk-ii-robotic-claw-with-servo-motor-1263193> (дата обращения: 18.02.2016).
13. Shabana A. A. Computational Dynamics, Third edition, John Wiley & Sons Inc., 2010.
14. URL: <http://electroprivod.ru/wg7152.htm> (дата обращения: 20.02.2016).

Simulation of the Actuator Dynamics of the Virtual Robots in the Training Complexes

E. V. Strashnov, strashnov_ev@mail.ru✉, **M. A. Torgashev**, mtorg@mail.ru,
Scientific Research Institute for System Analysis, RAS, Moscow, 117218, Russian Federation

Corresponding author: **Strashnov E. V.**, Leading Programmer,
Scientific Research Institute for System Analysis, RAS, Moscow, 117218, Russian Federation
e-mail: strashnov_ev@mail.ru

Received on May 25, 2016

Accepted on June 04, 2016

The topic of the article is the problem of the actuator dynamics simulation for the training complexes. Usually, when the mathematical model for a motor is formulated, its passport parameters are used: starting torque, idle speed and electromechanical time constant. The actuator creates the control action, which makes it possible to control the relative motion of the two joining rigid bodies. For the real-time dynamic simulation of the articulated rigid bodies the method of the sequential impulses is proposed. Within this method the actuator dynamics is realized by means of a special constraint, which correlates the bodies' coordinates with the engine's angle rotation. In order to model the complex mechanisms, which are controlled by a single motor, a special "repeater" constraint is proposed. This constraint allows us to simulate the movement of the manipulator gripper jaws. The method of the sequential impulses with stabilization on the basis of the split impulses is iterative, while the use of the specific criteria for the end of iteration allows a real-time simulation. The proposed algorithms and methods are implemented in the software modules written in C++. Their approbation was carried out in a dynamic subsystem for a robot simulator. The studies demonstrated that the proposed methods and algorithms allow us to solve the problems, associated with implementation of different kinds of robots control. Such technologies can also be used in the virtual labs, simulation complexes, systems of augmented virtual environment and other applications.

Keywords: actuator, articulated body system, actuator constraint and "repeater" constraint, sequential impulses method, virtual environment system

Acknowledgements: This work was supported by RFBR grant projects № 16-07-01104, № 16-37-00107 mol_a

For citation:

Strashnov E. V., Torgashev M. A. Simulation of the Actuator Dynamics of the Virtual Robots in the Training Complexes, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2016, vol. 17, no. 10, pp. 762–768.

DOI: 10.17587/mau.17.762-768

References

1. **Asada H.** Introduction to Robotics, *Lecture Notes*, 2005.
2. **Spong M. W., Hutchinson S., Vidyasagar M.** Robot Modeling and Control, New York, John Wiley & Sons, Inc., 2006.
3. **Tjagunov O. A.** *Matematicheskie modeli i algoritmy upravleniya promyshlennykh transportnykh robotov* (Mathematical Models and Control Algorithms of Industrial Transport Robots), *Informacionno-izmeritel'nye i upravljajushhie sistemy*, 2007, vol. 5, no. 5, pp. 63–69 (in Russian).
4. **Kardonov G. A.** *Jelektricheskie mashiny* (Electrical Machines), Lecture Notes, Sankt-Peterburgskij gosudarstvennyj institut tochnoj mehaniki i optiki, 2002, 145 p. (in Russian).
5. **Weinstein R., Guendelman E., Fedkiw R.** Impulse-Based Control of Joints and Muscles, *IEEE Transactions on Visualization and Computer Graphics*, 2008, vol. 14, no. 1, pp. 37–46.
6. **Kenny Erleben.** Stable, robust, and versatile multibody dynamics animation, PhD thesis, University of Copenhagen, 2004.
7. **Steffen Wittmeier.** Physics-Based Modeling and Simulation of Musculoskeletal Robots, PhD thesis, Technical University of Munich, 2014.
8. **Mihajljuk M. V., Strashnov E. V.** *Modelirovanie sistemy svyazannykh tel metodom posledovatel'nykh impul'sov* (Simulation of Articulated Multibody System Using Sequential Impulses Method), *Trudy NIISI RAN*, 2014, vol. 4, no. 2, pp. 52–60 (in Russian).
9. **Strashnov E. V., Mihajljuk M. V.** *Modelirovanie ogranichenij na otositel'noe dvizhenie sharnirno svyazannykh tel* (Simulation of Restrictions for the Relative Motion of the Articulated Rigid Bodies in the Virtual Environment Systems), *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2015, vol. 16, no. 10, pp. 678–685 (in Russian).
10. **Mihajljuk M. V., Strashnov E. V.** *Modelirovanie dinamiki sistemy svyazannykh tel s uchedom trenija v sharnirah* (Simulating Dynamics of the System of Articulated Rigid Bodies with Joint Friction), *Science & Education of the Bauman MSTU, Electronic journal*, 2016, no. 1, pp. 108–124 (in Russian).
11. **Kolesnikov K. S.** ed. *Kurs teoreticheskoy mehaniki: Uchebnik dlja vuzov* (Course of Theoretical Mechanics: A Textbook for high schools), Moscow, Publishing House of the Bauman MSTU, 2005, 736 p. (in Russian).
12. **DAGU MK II** claw with servo motor, available at: <http://www.electronshtik.ru/item/mk-ii-robotic-claw-with-servo-motor-1263193> [2016-18-02].
13. **Shabana A. A.** *Computational Dynamics*, Third edition, John Wiley & Sons Inc., 2010.
14. available at: <http://electroprivod.ru/wg7152.htm> [2016-20-02].

А. Н. Горитов, д-р техн. наук, проф., ang@asu.tusur.ru,
Томский Государственный университет систем управления и радиоэлектроники

Алгоритм синтеза промышленных роботов на основе заданного набора модулей

Рассматривается метод синтеза промышленных роботов, ориентированных на выполнение заданной технологической операции. Синтез выполняется на основе типового ряда унифицированных модулей, параметров выполняемой операции и параметров рабочей среды, в которой предполагается функционирование синтезированного робота. Рассмотрен алгоритм автоматизированного синтеза роботов на основе агрегатно-модульного метода построения роботов.

Ключевые слова: промышленные роботы, агрегаты, модули, синтез

Введение

Одним из перспективных направлений развития робототехники является агрегатно-модульный метод построения промышленных роботов (ПР) [1, 2]. Преимуществом этого метода является значительное сокращение времени на разработку и выпуск роботов по сравнению с традиционным подходом [3–5]. Кроме того, для агрегатно-модульных роботов повышается ремонтпригодность, а наличие модулей с различными энергомассогабаритными характеристиками позволяет получать роботы, имеющие различную стоимость и требующие различные эксплуатационные затраты.

Положительным фактором агрегатно-модульного подхода к построению роботов является тот факт, что становится возможным формализовать процесс проектирования робота. В связи с этим одной из актуальных задач является разработка методов и алгоритмов синтеза и анализа ПР на основе агрегатно-модульного подхода.

В работах [1, 6] рассмотрены общие алгоритмы синтеза и анализа агрегатно-модульных ПР. Базовой процедурой автоматизированного синтеза ПР является формирование структуры ПР. С помощью этого алгоритма разрабатываются различные варианты компоновки робота, которые должны обеспечить выполнение заданной технологической операции ПР. В общем случае этот алгоритм формирует множество вариантов проектируемого робота, которые будут анализироваться на последующих этапах. В работе предлагается рекурсивный алгоритм синтеза структуры роботов-манипуляторов при построении робота на основе агрегатно-модульного подхода. Вычислительная сложность предложенного алгоритма оценивается произведением числа звеньев робота на число различных типов модулей, используемых при синтезе роботов.

Решение задачи

Синтез робота, предназначенного для выполнения заданной операции, можно выполнить путем проектирования всех возможных вариантов роботов из заданного набора модулей и агрегатов с последующим анализом каждого из них. Такой подход

позволяет выбрать конструкцию робота, наиболее подходящую для выполнения заданной технологической операции, но требует больших ресурсных затрат. Так, синтез m -звенного робота с использованием n различных типов модулей без ограничений на соединение звеньев приведет к необходимости проанализировать $n(n-1)(n-2)\dots(n-m-1) = n! - m!$ вариантов. Наличие ограничений на типы используемых звеньев и приводов позволит сократить число рассматриваемых вариантов синтезируемого робота. Тем не менее, число вариантов, которые потребуются проанализировать, остается достаточно большим.

Сокращения ресурсных требований можно добиться путем использования методов, обеспечивающих исключение из рассмотрения вариантов, заведомо не удовлетворяющих заданным техническим требованиям, путем формирования набора правил, которые определяют условия, при которых допускается соединение модулей друг с другом.

Для описания предложенного метода ограничения одним классом синтезируемых роботов. Пусть синтезируемые роботы относятся к классу роботов-манипуляторов с разомкнутой кинематической цепью.

Рассмотрим алгоритм синтеза. Алгоритм основан на последовательном переборе модулей из числа доступных и контроле на соответствие синтезируемого робота техническому заданию.

Алгоритм СИНТЕЗ

Вход:

- техническое задание, содержащее технические характеристики требуемого робота;
- ограничения на синтезируемый вариант робота — число звеньев, типы кинематических узлов, особенности стыковки звеньев и т.д.;
- множество компонентов, которые могут использоваться при синтезе робота.

Выход:

- синтезированный вариант робота, удовлетворяющий заданным условиям.

1. Из заданного множества модулей — основной выбрать первый неиспользуемый модуль. Отметить модуль как используемый.

2. Проверить ограничения на первый модуль робота. Если модуль не удовлетворяет ограничениям, заданным в технических требованиях, то отметить, что модуль нельзя использовать в качестве первого, и вернуться на первый шаг. Иначе перейти на шаг 3.

3. Из заданного множества модулей выбрать неиспользуемый модуль. Отметить, что этот модуль уже используется. Перейти на шаг 4.

4. Проверить ограничения на включение этого модуля в синтезируемый робот. Если модуль не удовлетворяет условиям технического задания, то отметить, что модуль нельзя использовать, и вернуться на шаг 3. В противном случае перейти на шаг 5.

5. Проверить, достигнуты ли заданные параметры синтезируемого устройства. Если да, то перейти на шаг 6. Иначе сформировать вектор доступных модулей для включения в синтезируемый робот на этой итерации и перейти на шаг 3.

6. Если по техническому заданию робот должен включать хват, то выбрать из множества схватов тот, который соответствует техническому заданию.

Конец алгоритма.

Описание алгоритма

Синтез по предложенному алгоритму предполагает, что все данные об используемом алгоритме занесены в таблицы. Предполагается использование четырех основных таблиц — с данными об основаниях робота, с данными о звеньях, с данными о кинематических узлах и с данными о схватах.

В процессе синтеза робота для каждого нового элемента робота эти таблицы позволяют сформировать множество модулей, которые могут быть установлены на месте этого элемента. Например, на месте кинематического узла вращения может быть установлен любой кинематический узел вращения, информация о котором есть в таблице с данными о кинематических узлах. С каждым элементом сформированного множества модулей связывается атрибут, который может принимать три значения: "не выбран", "выбран" и "выбрать нельзя".

Синтез робота начинается с выбора основания робота. Выбранное основание помечается как "выбранное". Если дальнейший анализ покажет, что выбранное основание не удовлетворяет техническому заданию, то среди не выбранных оснований выбирается новое основание, а текущее отмечается как "выбрать нельзя".

После успешного выбора основания выполняется проверка технического задания и затем выбирается звено или кинематический узел. Выбранное звено или кинематический узел в соответствующем списке помечается как "выбран". Если в процессе дальнейшего анализа будет установлено несоответствие выбранного элемента техническому заданию или каким-либо ограничениям, то параметр выбора заменяется на "выбрать нельзя", а среди невыбранных элементов выполняется поиск нового элемента.

Процесс синтеза заканчивается после достижения заданных значений параметров.

Анализ вычислительной сложности алгоритма

Алгоритм начинает синтез робота с выбора основания. Затем, перебирая модули, выбирает среди них те, которые удовлетворяют техническим условиям. Синтезируемый робот постепенно наращивается до ограничений, заданных в техническом задании. Сложность алгоритма в этом случае составляет mn , где m — число звеньев робота, а n — число различных типов модулей.

Формализованное описание структуры синтезированного робота

Формально структура синтезируемого манипулятора будет описываться следующим образом:

[COMPONENT],

где COMPONENT — это пара, состоящая из имени элемента и списка присоединенных к нему элементов. Эта пара представлена следующим образом:

(COMP_NAME, NEXT_COMPONENTS),

где COMP_NAME — имя элемента, NEXT_COMPONENTS — список присоединенных элементов к данному. Список имеет следующий вид:

[C1, C2, ...],

где C_i — пара компонент—список, как описано выше. Список может быть пуст.

Компонентное представление исследуемого объекта

Для математического описания исследуемого объекта воспользуемся методом компонентных цепей. Компонентная цепь синтезируемого робота-манипулятора в этом случае представляется в виде дерева. Корень дерева — это специальный компонент, связанный с неподвижной системой координат. Узлы дерева — это компоненты, задающие звенья и кинематические узлы робота. Ветви дерева являются направленными (рис. 1). Корень дерева обозначен 0, узлы дерева — C1, C2, C3, C4, C5.

Формализованное представление синтезируемого устройства легко отображается в компонент-

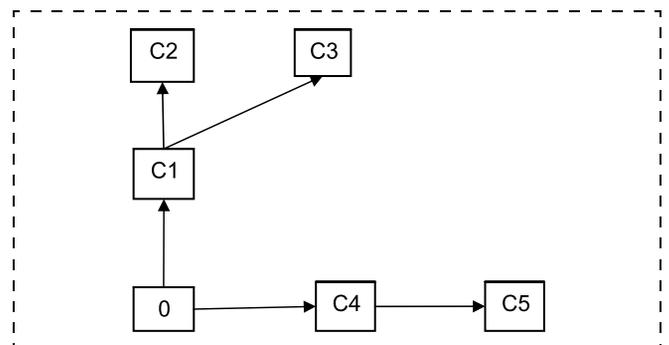


Рис. 1. Пример компонентной цепи

ное представление, которое в дальнейшем используется на этапе анализа синтезированного робота. Таким образом, компонентная цепь, представленная на рис. 1, в данном представлении будет выглядеть следующим образом:

$$\{(0, [(C1, [(C2, []), (C3, [])]), (C4, [(C5, [])])])\}$$

Здесь можно сделать следующие замечания:

1) компонент в начале цепи (начальный, первый компонент цепи) — компонент, который стоит в корне дерева;

2) компонент в конце цепи (конечный, последний компонент цепи) — компонент, который является листом дерева;

3) соседние компоненты в цепи — любые два компонента в любом пути от корня дерева к листьям, связанные одной ветвью (один такой путь для примера выделен на рис. 1, соседними в этом пути будут компоненты 0 и C1, C1 и C3).

Генерация компонентной цепи проводится с помощью алгоритма синтеза на основе информации о модулях и правилах их стыковки.

Исходные данные и ограничения

Исходные данные на проектирование робота берутся из технического задания. Минимальный набор требований, предъявляемых к проектируемому изделию, включает:

- число степеней подвижности;
- зону обслуживания робота;
- грузоподъемность;
- типы кинематических узлов.

При синтезе ПР на кинематическую цепь могут накладываться следующие условия и ограничения:

- 1) ограничение на длину кинематической цепи;
- 2) ограничение на тип модуля в начале цепи (не имеет смысла ставить в начале цепи кинематический или информационный модуль);
- 3) ограничения на тип модуля в конце цепи;
- 4) ограничение на число присоединенных модулей;
- 5) ограничения на соседние модули:
 - а) не ставить подряд два модуля одного типа;
 - б) не ставить подряд два одинаковых модуля.

Вышеописанные условия делятся на два типа: те, что проверяются непосредственно при генерации кинематической цепи (все, кроме третьего), и те, что проверяются после того, как цепь сгенерирована (третье условие из списка выше).

Анализ синтезированного робота

В процессе автоматизированного синтеза по заданному техническому заданию в общем случае может быть получено несколько вариантов роботов. Оценка и принятие решения по выбору окончательного варианта

синтезированного робота выполняется на основе результатов моделирования функционирования робота в предполагаемой внешней среде, включая расчет кинематических и динамических характеристик роботов при выполнении заданной операции.

Для этого синтезированный вариант робота конвертируется в компонентную цепь и передается в моделирующую программу. Моделирующая программа позволяет представить синтезированный вариант робота в трехмерном виде, задать внешнюю среду, в которой предполагается функционирование робота и оценить функционирование робота с учетом ограничений, накладываемых внешней средой [7, 8]. Для этого выполняется планирование траектории перемещения робота в сформированной внешней среде и расчет кинематических и динамических характеристик при выполнении заданной операции.

Пример

Рассмотрим пример работы модуля синтеза.

Для примера попытаемся сгенерировать роботоманипулятор, состоящий из трех звеньев и имеющий две степени свободы. В качестве рабочего органа (схвата) будем принимать компонент шар, в качестве остальных звеньев — цилиндр.

Условия синтеза

Основные требования к синтезируемому манипулятору задаются на главной форме проекта (рис. 2). Таким образом, имеем следующие условия синтеза:

- компоненты, участвующие в синтезе:
 - два твердотельных: цилиндр, шар;
 - один кинематический: вращение;
- число последующих компонентов за:
 - цилиндром: 1;
 - элементом вращения: 1;
 - шаром: 0;
- начинать и заканчивать цепь должны твердотельные компоненты;
- два подряд идущих элемента цепи не должны быть одинаковыми или одного типа;
- длина синтезированной цепи: не более пяти элементов.

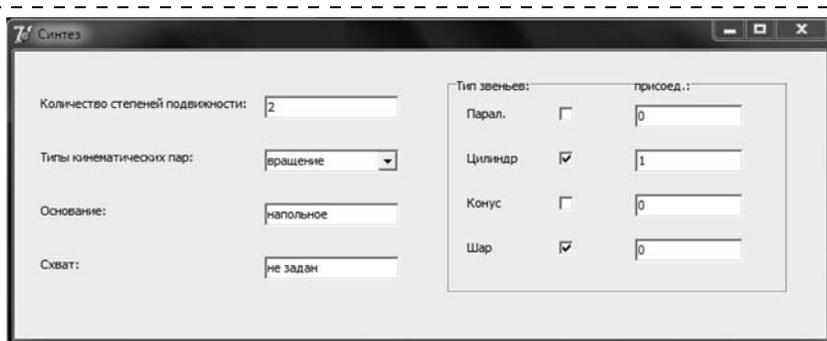


Рис. 2. Главная форма проекта

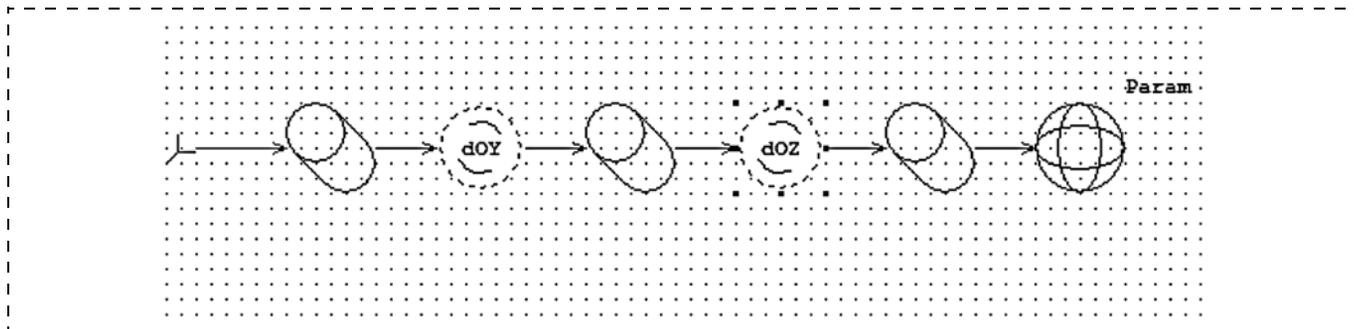


Рис. 3. Синтезированная цепь в структурном редакторе

Компонентная цепь

В результате были получены следующие конфигурации компонентных цепей:

- 1) [cSHAR,[]];
- 2) [cCYLINDER,[]];
- 3) [cCYLINDER,[cSHAR,[]]];
- 4) [cCYLINDER,[cFree_Rotate_Move,[cSHAR,[]]];
- 5) [cCYLINDER,[cFree_Rotate_Move,[cCYLINDER,[]]];
- 6) [cCYLINDER,[cFree_Rotate_Move,[cCYLINDER,[cSHAR,[]]]];
- 7) [cCYLINDER,[cFree_Rotate_Move,[cCYLINDER,[cFree_Rotate_Move,[cSHAR,[]]]]]];

- 8) [cCYLINDER,[cFree_Rotate_Move,[cCYLINDER,[cFree_Rotate_Move,[cCYLINDER,[]]]]]];
- 9) [cCYLINDER,[cFree_Rotate_Move,[cCYLINDER,[cFree_Rotate_Move,[cCYLINDER,[cSHAR,[]]]]]]]].

Видно, что они удовлетворяют условиям синтеза.

Очевидно, наиболее интересным является последний вариант.

Импортированная конфигурация

На рис. 3, 4 показан последний вариант синтезированного робота в окне редактора: структурного (рис. 3) и трехмерного (рис. 4).

После незначительной доработки в специализированном графическом редакторе модель робота принимает вид, показанный на рис. 5.

Дальнейший контроль результатов синтеза и оценка характеристик робота выполняется с помощью системы автоматизированного моделирования [7, 8], которая обеспечивает моделирование выполнения заданной технологической операции.

Заключение

В статье рассмотрена задача синтеза роботов на основе агрегатно-модульного метода построения роботов. Предложен алгоритм, который позволяет выполнить синтез устройства, удовлетворяющего заданным техническим требованиям. Сборка модулей в готовое изделие выполняется на основе

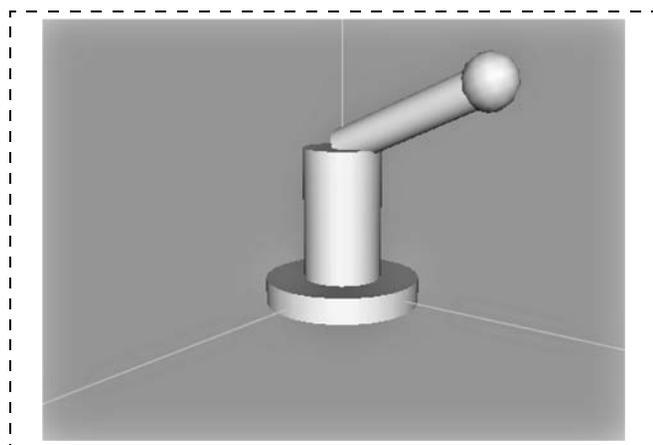


Рис. 4. Синтезированная модель в трехмерном редакторе

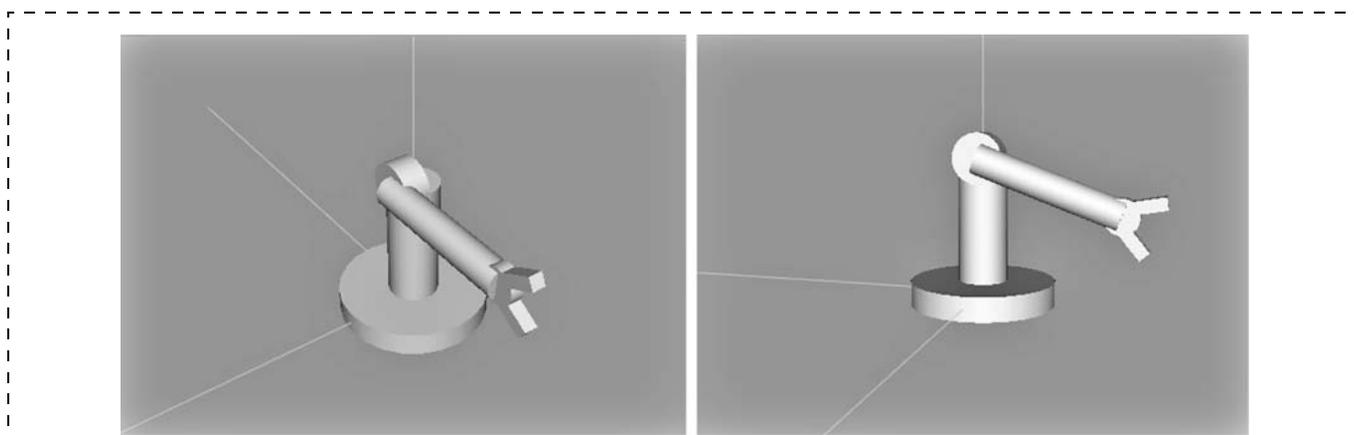


Рис. 5. Доработанная модель робота

правил стыковки модулей между собой. Вся информация о модулях хранится в базе данных.

Одним из важных моментов рассматриваемого метода синтеза является тот факт, что синтез робота выполняется с учетом параметров внешней среды и характера выполняемой операции.

Для оценки кинематических и динамических характеристик синтезированных вариантов роботов они передаются в моделирующую программу, где проводится расчет геометрических, кинематических и динамических характеристик синтезированных роботов и выполняется окончательный отбор робота.

Список литературы

1. **Воробьев Е. И., Козырев Ю. Г., Царенко В. И.** Промышленные роботы агрегатно-модульного типа. М.: Машиностроение, 1988. 240 с.

2. **Челпанов И. Б.** Устройства промышленных роботов. СПб.: Политехника, 2001. 203 с.

3. **Бурдаков С. Ф., Дъяченко В. А., Тимофеев А. Н.** Проектирование манипуляционных промышленных роботов и роботизированных комплексов: Учеб. пособ. для студ. вузов. М.: Высшая школа, 1986. 264 с.

4. **Khalil W., Dombre E.** Modeling, Identification and Control of Robots, 2004. 500 p.

5. **Zhao J., Feng Z., Chu F., Ma N.** Advanced Theory of Constraint and Motion Analysis for Robot Mechanisms. Academic Press, 2013. 496 p.

6. **Горитов А. Н.** Синтез и анализ агрегатно-модульных управляемых механических систем // Приборы и системы. Управление, контроль, диагностика. 2005. № 10. С. 10—14.

7. **Горитов А. Н.** Имитационное моделирование управляемой механической системы и ее рабочего пространства // Приборы и системы. Управление, контроль, диагностика. 2000. № 5. С. 11—13.

8. **Горитов А. Н.** Архитектура системы автоматизированного моделирования робототехнических комплексов // Программные продукты и системы: Приложение к журналу "Проблемы теории и практики управления". 2001. № 2, С. 17—19.

Synthesis Algorithm for Industrial Robots Based on a Given Set of Modules

A. N. Goritov, ang@asu.tusur.ru✉,

Tomsk State University of Control Systems and Radioelectronics, Tomsk, 634050, Russian Federation

Corresponding author: **Goritov Aleksandr N.**, D. Sc., Professor,
Tomsk State University of Control Systems and Radioelectronics, Tomsk, 634050, Russian Federation
e-mail: ang@asu.tusur.ru

Received on June 09, 2016

Accepted on June 27, 2016

One of the promising directions of robotics is the aggregate-modular method of construction of the industrial robots. The aggregate-modular method of construction of the industrial robots has many advantages. One of the positive factors of the aggregate-modular approach to construction of robots is a possibility to formalize the design process of robots. In this connection, one of the urgent tasks is development of the methods and algorithms for synthesis and analysis of the industrial robots on the basis of the aggregate-modular approach. The basic algorithm for an automated synthesis of the industrial robots is the synthesis of the structure of an industrial robot. The synthesis is performed on the basis of a model range of the standard modules, settings, operations and settings of the working environment, in which the functions of a robot are synthesized. The author proposes a recursive algorithm for a structure synthesis of the robot manipulators for building a robot based on the aggregate-modular approach. The proposed synthesis algorithm allows us to reduce the computing resources procedure required for the synthesis compared with a full search of all the possible variants and selection of the most suitable option of a robot. The results of the synthesis are converted into a format, readable by a dedicated system of simulations. This allows us to evaluate the synthesized version of the robot when performing scheduled operations in the external environment.

Keywords: industrial robots, design, units, modules, design automation, synthesis, algorithm

For citation:

Goritov A. N. Synthesis Algorithm for Industrial Robots Based on a Given Set of Modules, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2016, vol. 17, no. 11, pp. 769—773.

DOI: 10.17587/mau.17.769-773

References

1. **Vorob'ev E. I., Kozыrev Ju. G., Carenko V. I.** *Промышленные роботы агрегатно-модульного типа* (Industrial robots aggregate-modular type), Moscow, Mashinostroenie, 1988, 240 p. (in Russian).

2. **Челпанов И. Б.** *Устройства промышленных роботов* (The device for industrial robots), SPb., Politehnika, 2001, 203 p. (in Russian).

3. **Burdakov S. F., D#jachenko V. A., Timofeev A. N.** *Проектирование манипуляционных промышленных роботов и роботизированных комплексов* (Design of manipulation of industrial robots and robotic systems), Moscow, Vysshaja shkola, 1986, 264 p.

4. **Khalil W., Dombre E.** Modeling, Identification and Control of Robots, 2004, 500 p.

5. **Zhao J., Feng Z., Chu F., Ma N.** Advanced Theory of Constraint and Motion Analysis for Robot Mechanisms, Academic Press, 2013, 496 p. (in Russian).

6. **Goritov A. N.** *Sintez i analiz agregatno-modul'nyh upravlyaemyh mehanicheskikh sistem* (Synthesis and analysis of aggregate-modular controlled mechanical systems), *Pribory i Sistemy. Upravlenie, Kontrol', Diagnostika*, 2005, no. 10, pp. 10—14 (in Russian).

7. **Goritov A. N.** *Imitacionnoe modelirovanie upravlyaemoj mehanicheskoy sistemy i ejo rabocheho prostranstva* (Simulation-driven mechanical system and its workspace), *Pribory i Sistemy. Upravlenie, Kontrol', Diagnostika*, 2000, no. 5, pp. 11—13 (in Russian).

8. **Goritov A. N.** *Arhitektura sistemy avtomatizirovannogo modelirovanija robototekhnicheskikh kompleksov* (Architecture computer aided modeling of robotic systems), *Programmnye produkty i sistemy: Prilozhenie k zhurnalu "Problemy teorii i praktiki upravlenija"*, 2001, no. 2, pp. 17—19 (in Russian).

С. Л. Зенкевич, д-р физ.-мат. наук, проф., zenkev@bmstu.ru,
Н. К. Галустян, аспирант, narekgalustyan@gmail.com,
Научно-учебный центр "Робототехника" МГТУ им. Н. Э. Баумана

Децентрализованное управление группой квадрокоптеров

Решается задача децентрализованного управления группой автономных квадрокоптеров при роевом и строевом движении. Решение этой задачи включило в себя следующие этапы: разработка нелинейной динамической модели движения квадрокоптера и ее линеаризация, синтез алгоритма управления по траектории одинарным квадрокоптером, разработка алгоритма децентрализованного управления квадрокоптерами в мультиагентной системе, построение моделей и компьютерная апробация разработанных алгоритмов. Приведены результаты моделирования применения алгоритма управления группой квадрокоптеров.

Ключевые слова: мультиагентная система, квадрокоптеры, динамическая модель, управление по траектории, групповое движение, строевое движение, роевое движение, децентрализованное управление

Введение

Растущая популярность мультикоптеров в классе мини-БПЛА объясняется достоинствами этих винтокрылых аппаратов. Обладая всеми преимуществами летательных аппаратов вертолетного типа (возможностями зависания в пространстве, вертикального взлета и посадки, высокой маневренности как в открытом, так и в закрытых помещениях), мультикоптеры в отличие от вертолетов имеют простой и экономичный механизм (оси пропеллеров зафиксированы), не требующий сложного технического обслуживания.

Прогрессирующая популярность наглядно подтверждает глобальный рост рынка БПЛА как в исторической ретроперспективе, так и в прогнозе аналитических агентств. Интересным трендом является увеличение доли дронов гражданского назначения. Именно в этом секторе прогнозируется 19 % роста в год против 5 % беспилотников в военном секторе. Однако наиболее стремительный рост ожидается в сегменте мультикоптеров: 22 % в год. За 5 лет прогнозируется повышение спроса почти в 3 раза до 2,3 млрд долл.

В основе этого лежит расширение областей применения мультикоптеров, в первую очередь, в гражданском секторе. Сегодня мультикоптеры активно используются в кинопроизводстве и аэрофото съемке, в инспекции инфраструктурных объектов, в сельском хозяйстве. Масштабным примером их коммерческого применения является процесс доставки мультикоптерами малогабаритных грузов до клиентов всемирно известным интернет-ритейлером Amazon.

Идея мультиагентной системы (МАС) на базе мультикоптеров начала развиваться совсем недавно и имеет перспективы для широкого применения.

Это обусловлено преимуществами МАС: расширенное знание окружающей среды за счет коммуникации внутри группы, взаимозаменяемость агентов, способность коллективно выполнять более сложные задачи быстрее и надежнее. В природе мы наблюдаем массу примеров роевого (стаинового) поведения муравьев, рыб, птиц и других животных, что позволяет этим природным МАС эффективнее добывать пищу и выживать.

Анализ литературы показал, что много научных трудов посвящено системе управления движением одинарного квадрокоптера, которая может базироваться на линейно-квадратичных регуляторах [5], ПД или ПИД регуляторах [1–4], нечетких регуляторах [7] и нейросетевых регуляторах.

Тем не менее, можно отметить лишь несколько работ, направленных на создание и апробацию алгоритмов группового управления летающих агентов в трехмерном пространстве. В работе [4] профессора В. Кумара реализована и апробирована система управления группой из 20 квадрокоптеров. Однако в лаборатории действует система камер, а на летательных аппаратах установлены специальные маркеры, что позволяет с высокой точностью в любой момент определять положение в помещении. Кроме того, алгоритм управления является централизованным: управление квадрокоптерами происходит с центрального компьютера.

Следует отметить алгоритм децентрализованного управления группой квадрокоптеров на базе правил К. Рейнольдса [8]. Исследователи из Венгрии создали МАС из 10 полностью автономных квадрокоптеров, способных выполнять реальные коллективные задачи. В отличие от работы [4], квадрокоптеры получают данные о внешней среде лишь с помощью GPS-датчиков и обмениваются информацией о своем положении с соседними агентами

в зоне покрытия беспроводных радиомодулей связи. В упомянутой работе модель управления также была оптимизирована для реальной среды, включающей шумы, задержки сигналов и внешние воздействия, такие как ветер.

Интересный алгоритм управления квадрокоптером в МАС представлен в работе [9]. Модель управления на базе решения Нэша задачи о переговорах была апробирована в реальных условиях при групповом движении трех автономных квадрокоптеров в открытом пространстве. Одной из отличительных особенностей работы является сложная динамическая модель движения квадрокоптера, учитывающая ряд аэродинамических эффектов [6, 9].

В целом по принципу управления все подходы к управлению мультиагентными системами можно поделить на централизованные и децентрализованные [16]. Децентрализованное управление подразумевает отсутствие единого управляющего центра формирования координационных команд для каждого из элементов группы [12]. При этом групповое движение может быть "роевым" или строевым. Среди отечественных работ можно отметить статью [17], где решается строевая задача, при которой квадрокоптеры должны сохранять заданную геометрическую топологию (формацию).

В данной работе предлагается оригинальный способ децентрализованного управления квадрокоптерами в мультиагентной системе на базе модифицированных правил Рейнольдса в двух режимах полета: строевом и "роевом".

Динамическая модель движения квадрокоптера

Система координат и основные параметры движения. Движение квадрокоптера рассматривается с использованием двух систем координат: неподвижной системы координат (НСК), связанной с Землей, и подвижной системой координат (ПСК), связанной с квадрокоптером (рис. 1). Начало ПСК совпадает с центром масс квадрокоптера, ось X_B пересекает оси пропеллеров 1 и 3, и ось Y_B пересекает оси пропеллеров 2 и 4.

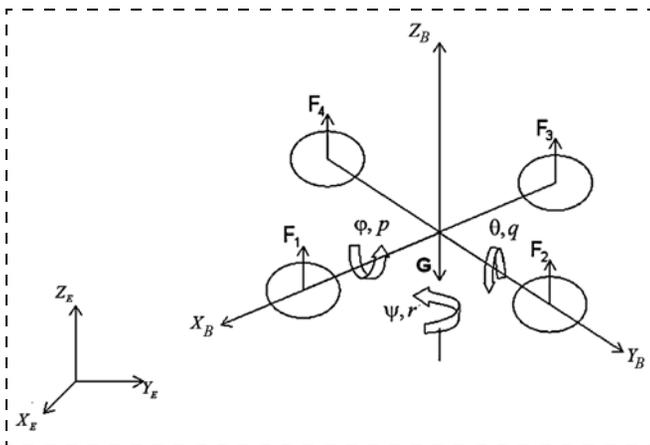


Рис. 1. Система координат и силы, действующие на квадрокоптер

Квадрокоптер имеет шесть степеней свободы: три угла и три линейные координаты центра масс. В движении квадрокоптера будем учитывать 12 параметров:

(φ, θ, ψ) — углы вращения;

$\Omega = (p \ q \ r)$ — вектор угловой скорости (проекции на оси ПСК);

$\mathbf{r} = (x \ y \ z)$ — вектор положения центра масс квадрокоптера;

$\dot{\mathbf{r}} = (\dot{x} \ \dot{y} \ \dot{z})$ — вектор линейной скорости.

Матрица поворота \mathbf{R}_{BE} между НСК и ПСК имеет следующий вид:

$$\mathbf{R}_{BE} = \begin{bmatrix} c_\psi c_\theta & c_\psi s_\theta s_\varphi - c_\varphi s_\psi & s_\psi s_\theta s_\varphi + c_\varphi c_\psi s_\theta \\ c_\theta s_\psi & c_\varphi c_\psi + s_\varphi s_\psi s_\theta & c_\varphi s_\psi s_\theta - c_\psi s_\varphi \\ -s_\theta & c_\theta s_\varphi & c_\varphi c_\theta \end{bmatrix},$$

где для компактности выражения используется следующее обозначение: $\sin \alpha = s_\alpha$, $\cos \alpha = c_\alpha$.

Силы и моменты, действующие на квадрокоптер.

Пропеллеры квадрокоптера вращаются со скоростью ω_i (i — номер пропеллера). При вращении пропеллеров возникают подъемные силы \mathbf{F}_i и моменты \mathbf{M}_i (аналогично несущему винту вертолета), вращающие квадрокоптер вокруг собственной оси (рис. 1), модули которых равны:

$$F_i = K_F \omega_i^2, \quad M_i = K_M \omega_i^2,$$

где K_F и K_M — константы. Следует отметить, что на практике динамические изменения пропеллеров происходят значительно быстрее динамических изменений твердого тела и аэродинамических изменений [4]. Следовательно, в качестве допущения при синтезе системы управления принимается, что угловые скорости вращения пропеллеров можно изменять моментально, и управление квадрокоптером осуществляется с помощью четырех силомоментных параметров $(u_1 \ u_2 \ u_3 \ u_4)^T = (F_{Z_B} \ M_{X_B} \ M_{Y_B} \ M_{Z_B})^T$:

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} K_F & K_F & K_F & K_F \\ 0 & K_F L & 0 & -K_F L \\ -K_F L & 0 & K_F L & 0 \\ K_M & -K_M & K_M & -K_M \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix}, \quad (1)$$

где L — расстояние между центром масс и осями пропеллеров.

Следующие аэродинамические эффекты не будут приниматься во внимание в силу сложности их учета и незначительности воздействия при невысоких скоростях полета [6]: изменение подъемной силы за счет дополнительного набегающего воздуха при полете квадрокоптера; изменение вектора подъемных сил за счет эффекта биения лопастей; сопротивление воздуха; ветер; гироскопический момент; эффект Земли при посадке и взлете.

Система уравнений движения квадрокоптера.

Вывод динамической модели движения одинарного квадрокоптера приведен в предыдущей работе [11]. Система уравнений движения получается из совокупности (i) уравнения второго закона Ньютона, (ii) уравнения, устанавливающего взаимосвязь вектора угловой скорости с матрицей поворота, и (iii) уравнения Эйлера. В векторном виде она имеет следующий вид:

$$\begin{cases} m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \mathbf{R}_B \mathbf{E} \begin{bmatrix} 0 \\ 0 \\ u_1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix}; \\ \dot{\hat{\Omega}} = \dot{\mathbf{R}} \mathbf{R}^T; \\ \mathbf{I}_B \dot{\Omega}_B = -\Omega_B \times \mathbf{I}_B \Omega_B + \begin{bmatrix} u_2 \\ u_3 \\ u_4 \end{bmatrix}, \end{cases} \quad (2)$$

где g — ускорение свободного падения; m — масса квадрокоптера; \mathbf{I}_B — тензор момента инерции; Ω_B — вектор угловой скорости в ПСК; $\dot{\Omega}_B$ — относительная производная вектора угловой скорости (производная в ПСК). После проекции на оси координат система уравнений движений примет вид

$$\begin{cases} \ddot{x} = u_1/m(s_\varphi s_\psi + c_\varphi c_\psi s_\theta); \\ \ddot{y} = u_1/m(c_\varphi s_\psi s_\theta - c_\psi s_\varphi); \\ \ddot{z} = u_1/m(c_\varphi c_\theta) - g; \\ \dot{\varphi} = p + (s_\varphi \operatorname{tg} \theta)q + (c_\varphi \operatorname{tg} \theta)r; \\ \dot{\theta} = c_\varphi q + (-s_\varphi)r; \\ \dot{\psi} = (s_\varphi/c_\theta)q + (c_\varphi/c_\theta)r; \\ \dot{p} = (u_2 - qr(I_{ZZ} - I_{YY}))/I_{XX}; \\ \dot{q} = (u_3 + pr(I_{ZZ} - I_{XX}))/I_{YY}; \\ \dot{r} = u_4/I_{ZZ}, \end{cases} \quad (3)$$

где I_{XX} , I_{YY} , I_{ZZ} — осевые моменты инерции.

Синтез алгоритма управления по траектории

Зададим траекторию квадрокоптера четырьмя параметрами — $[\mathbf{r}_T(t) \ \psi_T(t)] = [x_T(t) \ y_T(t) \ z_T(t) \ \psi_T(t)]$ (три координаты положения квадрокоптера и углом поворота относительно вертикальной оси (углы тангажа и крена будут определяться автоматически)).

Зададим три уровня управления.

1. Управление положением на базе ПД регулятора:

$$\begin{cases} \ddot{\mathbf{r}}_d = \ddot{\mathbf{r}}_T - \mathbf{K}_p \varepsilon_p - \mathbf{K}_d \varepsilon_d; \\ \varepsilon_p = (\mathbf{r} - \mathbf{r}_T); \\ \varepsilon_d = (\dot{\mathbf{r}} - \dot{\mathbf{r}}_T), \end{cases} \quad (4)$$

где \mathbf{K}_p и \mathbf{K}_d — положительно определенные матрицы усиления.

2. Управление ориентацией:

$$\begin{cases} [u_2 \ u_3 \ u_4]^T = -\mathbf{K}_R \mathbf{e}_R - \mathbf{K}_\Omega \mathbf{e}_\Omega; \\ \mathbf{e}_R = 1/2(\mathbf{R}_d^T \mathbf{R} - \mathbf{R}^T \mathbf{R}_d)^\vee; \\ \mathbf{e}_\Omega = \Omega - \Omega_T, \end{cases} \quad (5)$$

где \mathbf{R}^* — матрица поворота, рассчитанная на основе программных углов (8); \mathbf{R} — матрица поворота, рассчитанная на основе обратных связей по текущим углам $[\varphi \ \theta \ \psi]$; $^\vee$ — оператор перевода кососимметрической матрицы в \mathbb{R}^3 .

3. Управление высотой:

$$u_1 = m(\ddot{z}_d + g)/c_\varphi c_\theta. \quad (6)$$

Схема управления представлена на рис. 2. Подробное описание алгоритма приведено в статье [11].

Децентрализованное управление группой квадрокоптеров

Рассмотрим МАС из автономных квадрокоптеров, способных обмениваться с соседними агентами в зоне слышимости информацией о своем положении и угле рыскания (внутренняя коммуникация может быть технически реализована на базе радио-

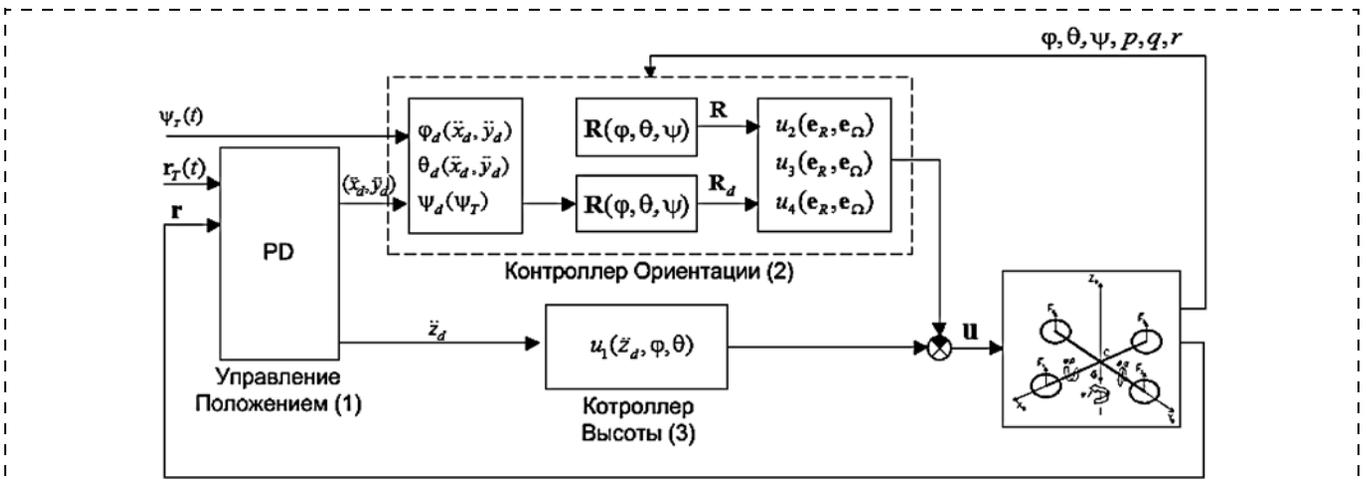


Рис. 2. Схема управления

модулей Xbee [8]). Вместе с этим, каждый квадрокоптер оснащен GPS- и IMU-датчиками для обратной связи по положению и угловой ориентации.

Требования к алгоритму. Зададим требования к алгоритму децентрализованного управления квадрокоптером в МАС. Во-первых, квадрокоптеры должны быть способны осуществлять групповой полет в трехмерном пространстве в двух режимах:

1) режим строевого движения — группа агентов стремится сохранить заданную геометрическую топологию (клин, сетка, цепочка) с двухуровневой иерархией — лидер и ведомые агенты;

2) режим "роевого движения" — коллективное движение происходит без необходимости выдерживать определенную геометрическую топологию, подобно стайному движению птиц.

К тому же, алгоритм должен соответствовать ряду критериев (см. таблицу).

Набор критериев для оценки качества группового движения

	Критерий	Определение
1	Безопасность	Агенты держатся на безопасном расстоянии друг от друга
2	Сплоченность	Двигаются на относительно близких и схожих дистанциях
3	Локальность	Агенты получают лишь локальную информацию в зоне слышимости, которая определяется дальностью связи
4	Масштабируемость	Алгоритм работает вне зависимости от числа агентов
5	Децентрализованность	Каждый агент имеет свою независимую систему управления
6	Взаимозаменяемость	Агенты взаимозаменяемы. Допускается потеря или приобретение новой связи

За основу алгоритма децентрализованного управления квадрокоптером в МАС были взяты плавающие Криса Рейнольдса.

Три правила Рейнольдса. Наблюдение за птицами вдохновило Рейнольдса на создание модели Birds для имитации поведения стаи птиц [10]. Он запрограммировал поведение каждого в отдельности агента на основе трех простых правил. Первое правило "Сплоченность" заключается в том, что агенты стараются держаться как можно ближе друг к другу. Второе правило "Разделение" обеспечивает, чтобы агенты стремились разойтись и сохранить безопасное расстояние друг от друга. Третье правило "Выравнивание скоростей" заключается в том, что агенты из одной группы стремятся двигаться с одинаковой скоростью (рис. 3, см. третью сторону обложки).

Четвертое правило. Модель Рейнольдса применима в случаях, когда допустимо "роевое" поведение группы. При задаче движения агентов с заданной геометрической топологией требуется модификация алгоритма. В требованиях мы указали, что группа автономных квадрокоптеров также должна быть способна двигаться строем. Добавим четвертое правило "Формация" для обеспечения

строевого движения, подобно системе из материальных точек, связанных упругими пружинами (рис. 4, см. третью сторону обложки). В такой модели при отклонении материальной точки возникают силы, возвращающие его в исходную позицию, сохраняя геометрическую топологию. Четвертое правило будет замещать правило "Сплоченности" при режиме полета строем.

Разработку децентрализованного алгоритма управления группой автономных квадрокоптеров в трехмерном пространстве осуществим эволюционным способом, начав с решения менее сложной плоской кинематической задачи.

Плоская кинематическая задача. Рассмотрим МАС, состоящую из идентичных агентов треугольной формы со следующим вектором состояния и управляющим воздействием:

$$\begin{cases} \mathbf{x} = [x \ y \ \varphi]; \\ \mathbf{u} = [V_X \ V_Y \ \omega], \end{cases}$$

где x и y — координаты агента, V_X и V_Y — проекции скорости агента, φ и ω — угол и угловая скорость вокруг собственной оси. Уравнение движения для агента имеет следующий вид:

$$\dot{\mathbf{x}} = \mathbf{u}.$$

Каждый агент получает информацию о положении других агентов, находящихся в его зоне слышимости. На основе этого агент регулирует скорость движения по модифицированному алгоритму Рейнольдса (рис. 5).

Управляющее воздействие по скорости для произвольного i -го агента зададим как сумму четырех компонент, соответствующих четырем правилам:

$$\begin{cases} u_1 = k_V V_{C_i,x} + k_C \sum_{|r_{ij}| < l_{\min}, |x_{ij}| < l_{\min}} \left(-\frac{x_{ij}}{|x_{ij}|} (l_{\min} - |x_{ij}|) \right) + \\ + k_{P_1} (1 - \gamma) r_{C_i,x} + k_{P_2} \gamma \varepsilon_{P_x}; \\ u_2 = k_V V_{C_i,y} + k_C \sum_{|r_{ij}| < l_{\min}, |y_{ij}| < l_{\min}} \left(-\frac{y_{ij}}{|y_{ij}|} (l_{\min} - |y_{ij}|) \right) + \\ + k_{P_1} (1 - \gamma) r_{C_i,y} + k_{P_2} \gamma \varepsilon_{P_y}; \\ u_3 = k_\varphi ((1 - \gamma)(\bar{\varphi} - \varphi_i) + \gamma \varepsilon_{P_\varphi}), \end{cases}$$

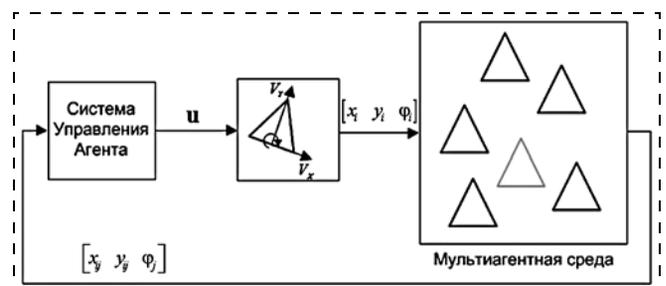


Рис. 5. Кинематическое управление агентом при двумерном групповом движении

где $k_V, k_C, k_{P_1}, k_{P_2}$ — коэффициенты усиления компонент, соответствующие порядку "Выравнивание скорости", "Разделение", "Сплоченность" и "Формация"; \mathbf{r}_{C_i} и \mathbf{V}_{C_i} — векторы положения и скорости центра масс группы агентов в зоне слышимости и системе координат i -го агента; x_{ij} и y_{ij} — проекции расстояния до соседнего j -го агента; l_{\min} — минимально безопасное расстояние между агентами; γ — режим полета (0 — роевое движение, 1 — строевое движение); ε_p — ошибка текущего положения агента относительно заданного. Способ управления угловой скоростью также зависит от режима полета: $\bar{\varphi}$ — среднее значение угла поворота среди агентов в зоне слышимости, ε_φ — ошибка текущего угла поворота агента относительно заданного. Заданная относительная позиция и угол поворота определяются выбранной геометрической топологией строя (например, клин, колонна, цепочка). Здесь $r_{C_i,x}, r_{C_i,y}, V_{C_i,x}, V_{C_i,y}, \varepsilon_{P_x}, \varepsilon_{P_y}$ — проекции соответствующих векторов в подвижной системе координат рассматриваемого i -го агента.

Примеры компьютерной апробации алгоритма можно найти по ссылке [21].

Плоская динамическая задача. Теперь рассмотрим МАС, состоящей из идентичных агентов треугольной формы со следующим вектором состояния и управляющим воздействием:

$$\begin{cases} \mathbf{x} = [x \ y \ \varphi \ V_x \ V_y \ \omega]; \\ \mathbf{u} = [F_x \ F_y \ L_z], \end{cases}$$

где F_x и F_y — проекции силы управления; L_z — момент управления. Уравнение движения агента будет иметь вид:

$$\begin{cases} \ddot{x} = (1/m)F_x; \\ \ddot{y} = (1/m)F_y; \\ \ddot{\varphi} = (1/I_{zz})L_z, \end{cases}$$

где m — масса агента, I_{zz} — момент инерции вокруг собственной оси вращения.

Управляющее воздействие по ускорению будем вычислять аналогичным образом:

$$\begin{cases} u_1 = m \left(k_V V_{C_i,x} + k_C \sum_{|r_{ij}| < l_{\min}, |x_{ij}| < l_{\min}} \left(-\frac{x_{ij}}{|x_{ij}|} (l_{\min} - |x_{ij}|) \right) + k_{P_1} (1 - \gamma) r_{C_i,x} + k_{P_2} \gamma \varepsilon_{P_x} \right); \\ u_2 = m \left(k_V V_{C_i,y} + k_C \sum_{|r_{ij}| < l_{\min}, |y_{ij}| < l_{\min}} \left(-\frac{y_{ij}}{|y_{ij}|} (l_{\min} - |y_{ij}|) \right) + k_{P_1} (1 - \gamma) r_{C_i,y} + k_{P_2} \gamma \varepsilon_{P_y} \right); \\ u_3 = m k_\varphi ((1 - \gamma)(\bar{\varphi} - \varphi_i) + \gamma \varepsilon_\varphi). \end{cases} \quad (7)$$

Каждое правило выражено в виде искусственной силы упругости. На рис. 6 представлена схема управления, где $\mathbf{P}_{2 \times q_N}^B = [r_{ij}^B]_{2 \times q_N}$ — положение соседних агентов в зоне слышимости в системе координат рассматриваемого агента.

Общая схема управления на базе модифицированных правил Рейнольдса представлена на рис. 7.

Примеры компьютерной апробации алгоритма можно найти по ссылке [22].

Задача группового управления квадрокоптерами в трехмерном пространстве. Рассмотрим МАС, состоящую из идентичных квадрокоптеров с управляющим воздействием (1), уравнением движения (2) и вектором состояния $\mathbf{x} = [z \ y \ z \ \dot{x} \ \dot{y} \ \dot{z} \ \varphi \ \theta \ \psi \ p \ q \ r]$.

Оператор выбирает режим поведения квадрокоптера (рис. 8): при $\gamma = 1$ включается режим лидера, тогда управление по скорости или по траектории выполняется оператором; при $\gamma = 0$ включается режим ведомого, тогда агент полностью автономен и регулирует свое движение в соответствии с мо-

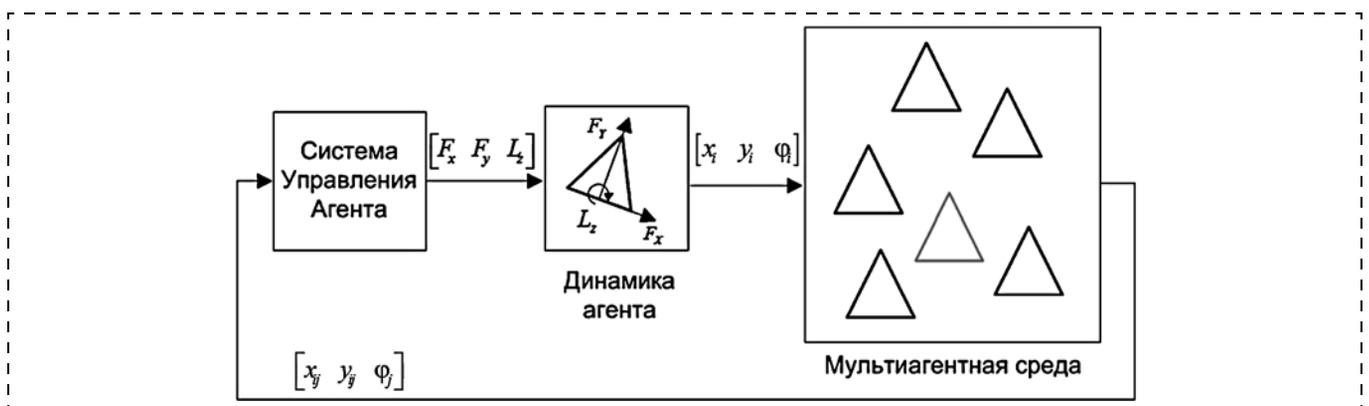


Рис. 6. Динамическое управление агентом при двумерном групповом движении

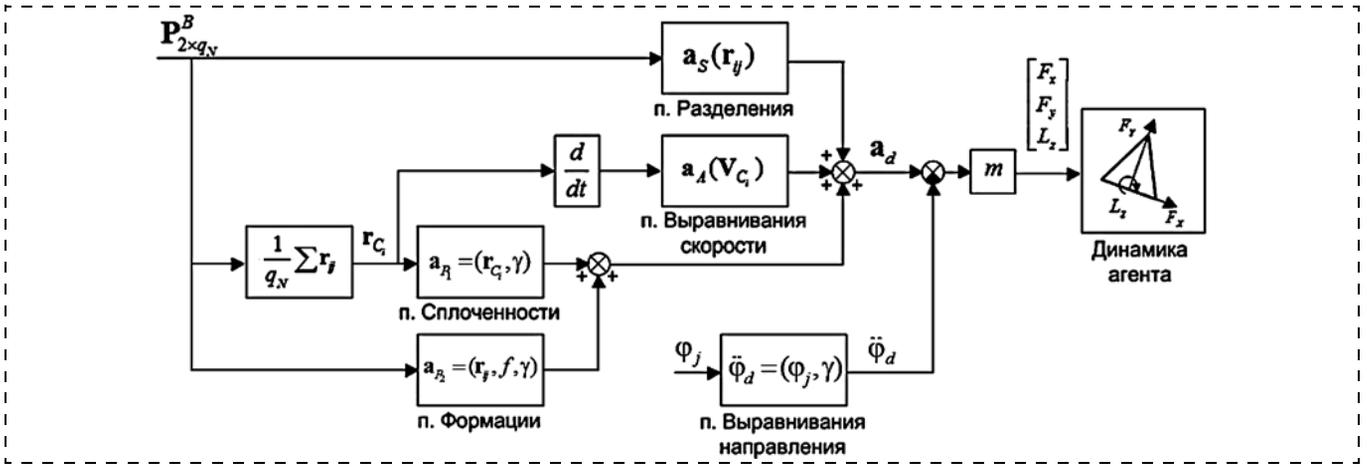


Рис. 7. Схема управления по ускорению на базе модифицированных правил Рейнольдса

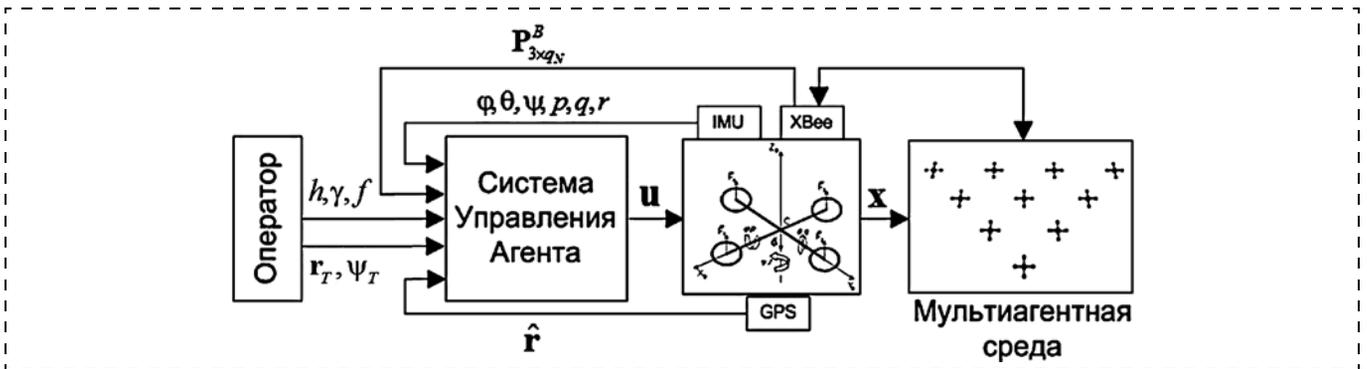


Рис. 8. Взаимодействие квадрокоптера с мультиагентной средой

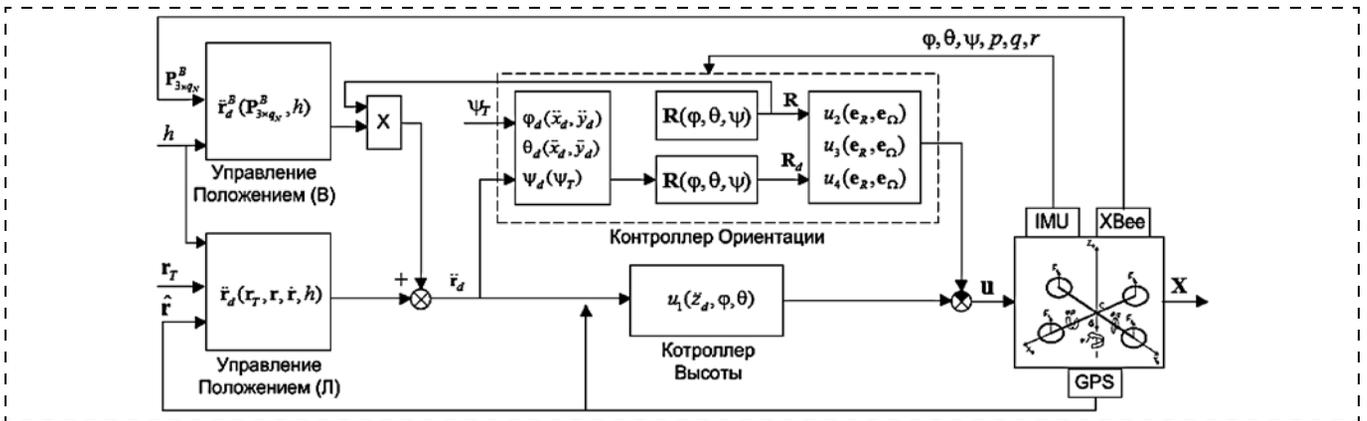


Рис. 9. Схема управления квадрокоптером в МАС

дифицированными правилами Рейнольдса, состоянием мультиагентной среды и режимом движения группы. Оператор также выбирает режим полета группы h (строевое или роевое) и номер заранее заданных форм строя f .

В основе алгоритма управления квадрокоптером в МАС лежат ранее разработанные модули управления одинарным квадрокоптером (4)–(6) и

модуль управления агентом в МАС в плоской динамической задаче (7).

Квадрокоптер в зависимости от режима поведения переключает блок управления положением. Если квадрокоптер является лидером, то включен блок управления положением (Л) (рис. 9), где выходной сигнал формируется согласно системе (4).

Если квадрокоптер является ведомым, то включен блок управления положением (В), и желаемое

ускорение формируется по аналогии с системой (7) с корректировкой на трехмерное пространство:

$$\begin{cases} \ddot{\mathbf{r}}_d^B = \mathbf{a}_V + \mathbf{a}_C + \mathbf{a}_{P_1} + \mathbf{a}_{P_2} + \mathbf{K}_d \varepsilon_d; \\ \mathbf{a}_V = \mathbf{K}_V \mathbf{V}_C; \\ \mathbf{a}_C = \mathbf{K}_C \sum_{|r_{ij}| < l_{\min}} \left(-\frac{\mathbf{r}_{ij}}{|r_{ij}|} (l_{\min} - |r_{ij}|) \right); \\ \mathbf{a}_{P_1} = \mathbf{K}_{P_1} (1 - \gamma) \mathbf{r}_C; \\ \mathbf{a}_{P_2} = \mathbf{K}_{P_2} \gamma \varepsilon_P. \end{cases} \quad (8)$$

Компьютерная апробация алгоритма управления по траектории

Для апробации алгоритма децентрализованного управления квадрокоптером в MAC использовалась связка (рис. 10) программных пакетов MATLAB и Universal Mechanism (UM) [20]. Квадрокоптеры в UM задаются как система из идентичных тел с одинаковыми масс-инерционными характеристиками. Для разработки модели системы управления использовали пакет MATLAB, у которого была установлена двухсторонняя связь с UM для осу-

ществления динамического и кинематического моделирования. Каждый агент имеет свою независимую систему управления в MATLAB. При этом из UM в MATLAB передаются такие параметры, как координаты, скорость и угловые параметры агента, а также координаты и угол рыскания соседних агентов в ЛСК искомого агента. В MATLAB проводится расчет управляющего воздействия (1) и передается в UM (рис. 10).

Для компьютерного моделирования использовали единый существующий прототип квадрокоптера со следующими масс-инерционными и геометрическими характеристиками:

$$\mathbf{I} = \begin{bmatrix} 0,002352 & 0 & 0 \\ 0 & 0,002352 & 0 \\ 0 & 0 & 0,004704 \end{bmatrix};$$

$m = 0,32 \text{ кг}; L = 0,209 \text{ м}.$

Как уже было отмечено ранее, модель системы управления была построена в MATLAB, где управляющие сигналы генерируются на основе формул (4)–(6) и (7). Матрицы усиления подобраны экспериментально и равны

$$\mathbf{K}_p = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}; \mathbf{K}_d = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}; \mathbf{K}_R = \begin{bmatrix} 20 & 0 & 0 \\ 0 & 20 & 0 \\ 0 & 0 & 20 \end{bmatrix};$$

$$\mathbf{K}_\Omega = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 5 \end{bmatrix}; \mathbf{K}_V = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}; \mathbf{K}_C = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 4 \end{bmatrix};$$

$$\mathbf{K}_{P_1} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}; \mathbf{K}_{P_2} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}; l_{\min} = 1.$$

На графиках на рис. 11 представлены результаты моделирования группового движения квадрокоптеров в режиме полета строем. Лидеру задана плоская синусоидальная траектория:

$$\begin{cases} x_L = 5 \sin(0,5t); \\ y_L = 0,5t; \\ z_L = 0. \end{cases}$$

Лидер выполняет плоское движение при средней скорости 3,5 м/с и описывает синусоиду вдоль оси X. Группа из 10 квадрокоптеров демонстрирует сплоченное коллективное движение: среднearифметическое расстояние между агентами колеблется около $\bar{d}_{ij} = 1,8 \text{ м}$ при стандартном отклонении в пределах $\sigma = 1 \text{ м}$. Сближение агентов происходит на поворотах.

Критерий безопасности также соблюдается: минимальная дистанция между парой агентов $\text{mind}_{ij} > 1,4 \text{ м}$.

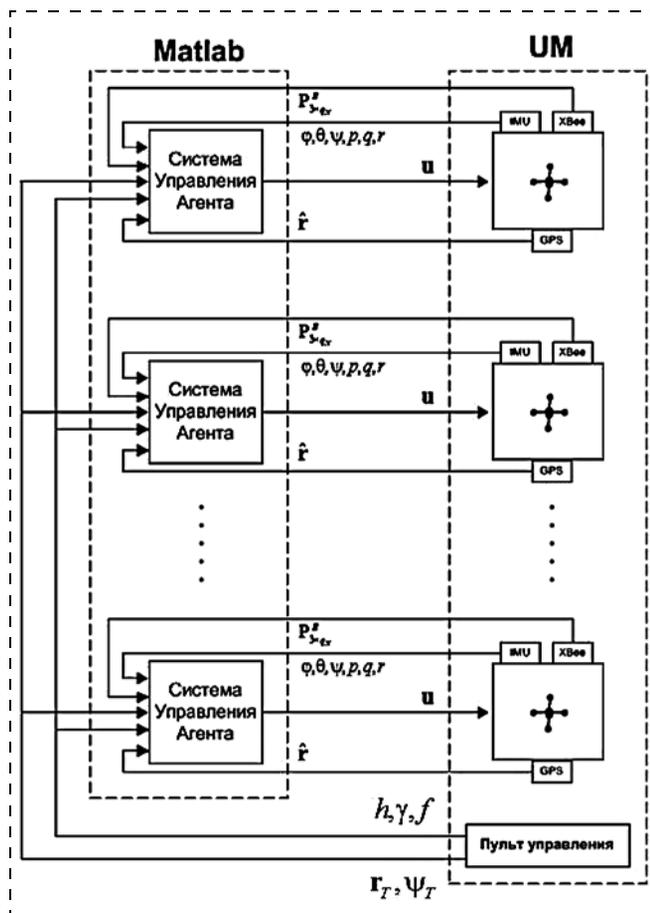


Рис. 10. Схема подключения "Matlab" и "Universal Mechanism"

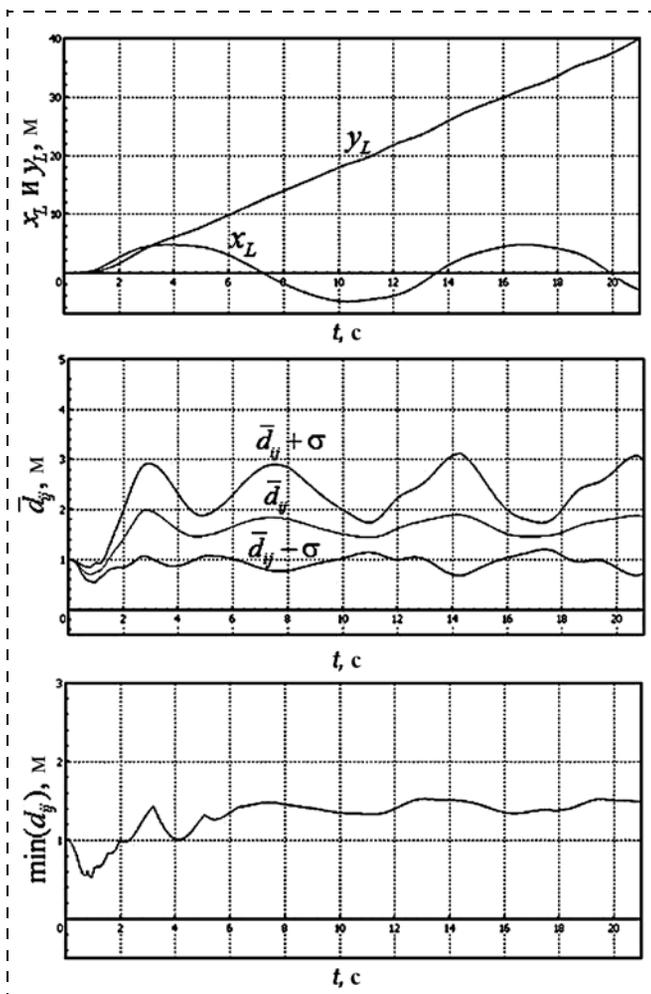


Рис. 11. Оценка критериев сплоченности и безопасности группового движения квадрокоптеров

Примеры компьютерной апробации алгоритма можно найти по ссылке [23].

Заключение

В работе получена динамическая модель движения одиночного квадрокоптера и алгоритм управления по траектории. На базе модифицированных правил Рейнольдса разработан децентрализованный алгоритм управления квадрокоптеров в мультиагентной системе для полета в двух режимах в трехмерном пространстве, отвечающий шести ключевым критериям согласованного и безопасно-го движения в группе.

С помощью компьютерного моделирования верифицирована работоспособность алгоритма управления квадрокоптерами в МАС.

Список литературы

1. Cutler M., How J. P. Actuator Constrained Trajectory Generation and Control for Variable-Pitch Quadrotors // AIAA Guidance,

Navigation, and Control Conference (GNC). Minneapolis, Minnesota, August 2012.

2. Lee T., Leok M., McClamroch N. H. Geometric Tracking Control of a Quadrotor UAV on SE(3) // 49th IEEE Conference on Decision and Control. 2011.

3. Mellinger D., Kumar V. Minimum Snap Trajectory Generation and Control for Quadrotors. GRASP Lab., University of Pennsylvania, 2011.

4. Kushleyev A., Mellinger D., Kumar V. Towards A Swarm of Agile Micro Quadrotors. GRASP Lab, University of Pennsylvania, 2013.

5. Thorhallur Tomas Buchholz, Dagur Gretarsson. Construction of a Four Rotor Helicopter Control System // S. M. Thesis. Technical University of Denmark, February 2009.

6. Hoffman G. M., Huang H., Waslander S. L., Tomlin C. J. Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment // AIAA Guidance, Navigation and Control Conference and Exhibit, August 2007, Hilton Head, South Carolina.

7. Hong S. K. Fuzzy logic based closed-loop strap down attitude system for unmanned aerial vehicle (UAV). Department of Aerospace Engineering, Sejong University, 2005.

8. Vasarhelyi G., Virágh Cs., Somorjai G., Tarcai N., Szdrényi T., Nepusz T., Vicsek T. Outdoor flocking and formation flight with autonomous aerial robots. Department of Biological Physics, Eotvos University, Hungary, 2014.

9. Hoffmann G., Huang H., Waslander S., Tomlin C. Precision flight control for a multi-vehicle quadrotor helicopter testbed // Control engineering practice. 2011. Vol. 19 (9). P. 1023–1036.

10. Reynolds C. Flocks, birds, and schools: A distributed behavioural model // Comput. Graph. 1987. Vol. 21, N. 4. P. 25–34.

11. Зенкевич С. Л., Галустян Н. К. Синтез и апробация алгоритма управления движением квадрокоптера по траектории // Мехатроника, автоматизация, управление. 2015. № 8. С. 530–535.

12. Морозова Н. С. Управление движением строя для мультиагентной системы, моделирующей автономных роботов // Вестник московского университета. Сер. 15. Вычислительная математика и кибернетика. 2015. № 4. С. 23–31.

13. Иванов Д. Я. Построение формаций в группах квадрокоптеров с использованием виртуального строя // XII Всероссийское совещание по проблемам управления ВСПУ. Москва, 2014.

14. Зенкевич С. Л., Ющенко А. С. Основы управления манипуляционными роботами. М.: Изд-во МГТУ, 2004.

15. Голубев Ю. Ф. Основы теоретической механики. М.: Изд-во МГУ, 2000.

16. Каляев И. А., Гайдук А. Р., Капустян С. Г. Модели и алгоритмы коллективного управления в группах роботов. М.: Физматлит, 2009.

17. Иванов В. А., Медведев В. С. Математические основы теории оптимального и логического управления. М.: Изд-во МГТУ, 2011.

18. Бесекерский В. А., Попов Е. П. Теория систем автоматического управления. СПб.: Профессия, 2007.

19. Белоконов С. А., Золотухин Ю. Н., Мальцев А. С., Нестеров А. А., Филиппов М. Н. Управление параметрами полета квадрокоптера при движении по заданной траектории // Автоматика. 2012. № 5. С. 32–41.

20. Universal Mechanism — the software for modeling of dynamics. URL: <http://www.universalmechanism.com>

21. Примеры компьютерной апробации алгоритма для плоской кинематической задачи. URL: www.youtube.com/watch?v=TM123zxlNso, www.youtube.com/watch?v=1kr6sEvJ6Ek

22. Примеры компьютерной апробации алгоритма для плоской динамической задачи. URL: <https://www.youtube.com/watch?v=yuidxOUm-o8>

23. Примеры компьютерной апробации алгоритма управления квадрокоптером в МАС. URL: www.youtube.com/watch?v=8a3Bgg3Vas, www.youtube.com/watch?v=aORTgg11aNk

Decentralized Control of a Quadcopter Swarm

S. L. Zenkevich, zenkev@bmstu.ru, N. K. Galustyan, narekgalustyan@gmail.com✉,
Robotic Center for Education and Research at Bauman Moscow State Technical University

Corresponding author: Galustyan N. K., Postgraduate Student,
Bauman Moscow State Technical University, Moscow, 105005, Russian Federation,
e-mail: narekgalustyan@gmail.com

Received on July 05, 2016

Accepted on July 20, 2016

In this article the authors propose a decentralized control algorithm for a swarm consisted of quadcopters, which are unmanned aerial vehicles (UAV) lifted and propelled by four rotors. The first step was derivation of a quadcopter flight dynamic (math) model and its further linearization. The math model was derived on the basis of Newton and Euler equation of motion. The next step was to design a trajectory control algorithm by using PD regulator for the trajectory error minimization. After designing and approbation of the control algorithm of a unit quadcopter, the authors proceeded further with the task of a group control. Originally, the authors set several requirements to the control algorithm of the quadcopter for a group flight, e.g. cohesion and safety of the flights, decentralization and scalability of the control algorithm. As a ground algorithm they selected C. Reynolds rules designed for "boids". In order to make it possible to switch a quadcopter group flight from the swarm mode to the formation mode, in addition to Reynolds rules the authors developed the forth rule called "Formation". Before designing of the decentralized control for a quadcopter swarm, the authors solved similar tasks related to the kinematic control and dynamic control of the agents moving in 2D. The authors set several experiments in a "Universal Mechanism" and "Matlab" of the quadcopter group flight of 10 agents with independent control systems. The above-mentioned experiments proved the efficiency of the designed decentralized control algorithm. The modeling details and experiment results are also presented in the paper.

Keywords: Swarm, multiagent system, flocking, quadcopter, dynamics, attitude control, trajectory control, formation flight, outdoor flocking, decentralized control, PD regulator, MATLAB, universal mechanism

For citation:

Zenkevich S. L., Galustyan N. K. Decentralized Control of a Quadcopter Swarm, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2016, vol. 17, no. 9, pp. 774—782.

DOI: 10.17587/mau.17.774-782

References

1. Cutler M., How J. P. Actuator Constrained Trajectory Generation and Control for Variable-Pitch Quadrotors, *AIAA Guidance, Navigation, and Control Conference (GNC)*, Minneapolis, Minnesota, August 2012.
2. Lee T., Leok M., McClamroch N. H. Geometric Tracking Control of a Quadrotor UAV on SE(3), *49th IEEE Conference on Decision and Control*, 2011.
3. Mellinger D., Kumar V. Minimum Snap Trajectory Generation and Control for Quadrotors, *GRASP Lab., University of Pennsylvania*, 2011.
4. Kushleyev A., Mellinger D., Kumar V. Towards A Swarm of Agile Micro Quadrotors, *GRASP Lab, University of Pennsylvania*, 2013.
5. Thorhallur Tomas Buchholz, Dagur Gretarsson. Construction of a Four Rotor Helicopter Control System, *S. M. Thesis, Technical University of Denmark*, 2009.
6. Hoffman G. M., Huang H., Waslander S. L., Tomlin C. J. Quadcopter Helicopter Flight Dynamics and Control: Theory and Experiment, *AIAA Guidance, Navigation and Control Conference and Exhibit, Hilton Head, South Carolina*, 2007.
7. Hong S. K. Fuzzy logic based closed-loop strap down attitude system for unmanned aerial vehicle (UAV), *Department of Aerospace Engineering, Sejong University*, 2005.
8. Vasarhelyi G., Virágh Cs., Somorjai G., Tarcai N., Szörényi T., Nepusz T., Vicsek T. Outdoor flocking and formation flight with autonomous aerial robots, *Department of Biological Physics, Eotvos University, Hungary*, 2014.
9. Hoffmann G., Huang H., Waslander S., Tomlin C. Precision flight control for a multi-vehicle quadrotor helicopter testbed, *Control engineering practice*, 2011, no. 19 (9), pp. 1023—1036.
10. Reynolds C. Flocks, birds, and schools: A distributed behavioral model, *Comput. Graph.*, 1987, vol. 21, no. 4, pp. 25—34.
11. Zenkevich S. L., Galustyan N. K., *Sintez i aprobatsiya algoritma upravleniya dvizheniem kvadroptera po traektorii* (Quadcopter trajectory control design and flight modeling), *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2015, no. 8, pp. 530—535 (in Russian).
12. Morozova N. S., *Upravlenie dvizheniem stroja dlja mul'tiagentnoj sistemy, modelirujushhej avtonomnyh robotov* (Formation control of multiagent group of autonomous robots), *Vestnik MSU, Vychislitel'naja Matematika i Kibernetika*, 2015, no. 4, pp. 23—31 (in Russian).
13. Ivanov D. Ya., *Postroenie formacij v gruppah kvadropteroev s ispol'zovaniem virtual'nogo stroja* (Control of quadcopter groups based on virtual formation), *XII Vserossijskoe Soveshhanie po Problemam Upravlenija VSPU, Moscow*, 2014 (in Russian).
14. Zenkevich S., Yuschenko A., *Osnovy upravlenija manipulacionnymi robotami* (Basics of manipulator control), *BMSTU*, 2004 (in Russian).
15. Golubev Ju., *Osnovy teoreticheskoy mehaniki* (Basics of theoretical mechanics), *MSU*, 2000 (in Russian).
16. Kaljaev I. A., Gajduk A. R., Kapustjan S. G., *Modeli i algoritmy kollektivnogo upravlenija v gruppah robotov* (Models and algorithms of collective control in robot groups), *Moscow, Fizmatlit*, 2009 (in Russian).
17. Ivanov V., Medvedev V., *Matematicheskie osnovy teorii optimal'nogo i logicheskogo upravlenija* (Mathematical basics of optimal and logic control theory), *BMSTU*, 2011 (in Russian).
18. Besekerskiy V., Popov E., *Teoriya sistem avtomaticheskogo upravlenija* (Theory of autonomous control), *Professija*, 2007 (in Russian).
19. Belokon' S., Zolotuhin Ju., Mal'cev A. S., Nesterov A. A., Filippov M. N., *Upravlenie parametrami poljota kvadroptera pri dvizhenii po zadannoju traektorii* (Control of flight main parameters in case of trajectory control), 2012, no. 5, pp. 32—41 (in Russian).
20. **Universal Mechanism**, the software for modeling of dynamics, available at: <http://www.universalmecanism.com>
21. **Link** on computer experiment of group control for 2D kinematic task, available at: www.youtube.com/watch?v=TMI23xINso, www.youtube.com/watch?v=lkr6sEvJ6Ek
22. **Link** on computer experiment of group control for 2D dynamic task, available at: <https://www.youtube.com/watch?v=yiidxOUm-o8>
23. **Link** on computer experiment of quadcopter group control, available at: www.youtube.com/watch?v=8a3Bpgg3Vas, www.youtube.com/watch?v=aORTgg11aNk

Г. Н. Лебедев, д-р техн. наук, проф., kaf301@mai.ru,

В. И. Гончаренко, д-р техн. наук, доц., директор военного института МАИ,

А. В. Румакина, ассистент, dolgova-221@mail.ru,

Московский авиационный институт (национальный исследовательский университет)

Модификация метода ветвей и границ для двумерной маршрутизации координированного полета группы летательных аппаратов¹

Предложен метод решения задачи коммивояжера в случае двумерной маршрутизации полета, отличающийся формированием двух исходных матриц расстояний между пунктами для двух разомкнутых маршрутов, которые анализируются поочередно при координированном выборе элемента минимальной длины в каждой из них. Приводятся примеры расчетов, демонстрирующие эффективность предложенного подхода и указывающие на сокращение числа пунктов облета более, чем в два раза.

Ключевые слова: беспилотный летательный аппарат, многомерная маршрутизация, безопасность полета, целочисленное программирование

Введение

В настоящее время при решении различных задач управления и контроля все больше внимания уделяется групповым действиям в полете, отличающимся высокой эффективностью. К числу таких задач относится обслуживание заданного множества пунктов наблюдения наземных объектов с помощью группы летательных аппаратов (ЛА), маршруты движения которых необходимо заранее определить.

Задача маршрутизации движения рассматривается во множестве работ, посвященных выбору оптимальных, но одномерных маршрутов [1–16], среди которых особое место занимает метод ветвей и границ [2] в виде решения известной математической задачи коммивояжера. Однако при групповых действиях приходится либо учитывать штрафы при пересечении одномерных маршрутов полета [3], либо осуществлять дополнительный прогнозируемый контроль безопасности полета при сближении ЛА, внося существенные поправки в первоначальные планы одномерной маршрутизации.

В данной работе рассматривается задача координированного маршрутного полета двух ЛА при следующей постановке задачи.

Постановка задачи

1. Решается задача формирования двух разомкнутых маршрутов движения, имеющих заданные общие начальную и конечную точки A и B (рис. 1).

2. Задано множество неподвижных точечных пунктов наблюдения, координаты их расположения и общее число n (на рис. 1 это число $n = 5$), что позволяет сформировать исходящую матрицу M_0 расстояний между ними, началом и концом $((n + 1)$ -й

точкой) маршрутов. Для рис. 1 исходная матрица M_0 имеет следующий вид:

	0	1	2	3	4	5	6
0	X	5	8	11	9,4	11,5	X
1	X	X	3	6	5,8	6,7	11
2	X	3	X	3	5	4,3	8
3	X	6	3	X	5,8	3	5
4	X	5,8	5	5,8	X	3,5	9,4
5	X	6,7	4,3	3	3,5	X	5,8
6	X	X	X	X	X	X	X

В этой матрице есть запрещенные элементы (обозначенные X), как это принято в известном методе ветвей и границ [6], а именно

- элементы по диагонали матрицы, запрещающие попадание в любой пункт повторно сразу после его обслуживания;
- элементы в первом столбце, запрещающие возвращение в начальный пункт для разомкнутых маршрутов;
- элементы в нижнем столбце, запрещающие любое продолжение движения после попадания в конечный пункт $(n + 1)$;
- элементы $(0, n + 1)$, запрещающие движение сразу из начального пункта в конечный.

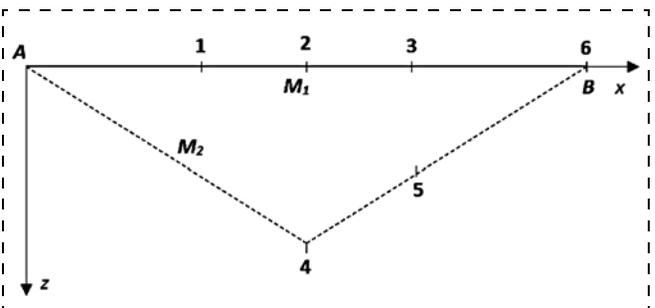


Рис. 1. Картина расположения пунктов наблюдения с указанием найденных в конце расчета маршрутов M_1 и M_2

¹ Работа выполнена при материальной поддержке грантов РФФИ 16-08-00-832-а, 15-08-00043.

3. Имеются два ЛА, полет которых на заданной постоянной высоте нужно скоординировать таким образом, чтобы каждый пункт был обслужен только одним ЛА и только один раз.

4. Требуется сформировать два маршрута полета так, чтобы их суммарная длина была минимальна.

Подход к решению задачи на первом этапе расчетов

Решение задачи основано на известных действиях, принятых в целочисленном методе ветвей и границ для одномерной маршрутизации, с добавлением к ним новых необходимых операций. Как и в известном методе, планирование должно состоять из множества шагов расчета ($k = 1, \dots, n + 1$) с тем отличием, что вместо одной матрицы M_0 расстояний анализируются две матрицы M_1 и M_2 — по одной для каждого маршрута. Исходное состояние этих матриц формируется на основе общей исходной матрицы (как показано ниже) присоединением нового столбца для указания координаты x и z так называемых "центров притяжения" обоих формируемых маршрутов, имеющих номера $(n + 2)$ и $(n + 3)$.

В исходном состоянии эти матрицы для примера, представленного на рис. 1, имеют вид:

▲ матрица M_{10} для первого маршрута на нулевом шаге

	0	1	2	3	4	5	6	$7 = n + 1$	
0	X	5	8	11	9,4	11,5	X	8	→ $X_{n+2}(0)$
1	X	X	3	6	5,8	6,7	11		
2	X	3	X	3	5	4,3	8		
3	X	6	3	X	5,8	3	5		
4	X	5,8	5	5,8	X	3,5	9,4		
5	X	6,7	4,3	3	3,5	X	5,8		
6	X	X	X	X	X	X	X	0	← $Z_{n+2}(0)$;

▲ матрица M_{20} для второго маршрута на нулевом шаге

	0	1	2	3	4	5	6	$7 = n + 1$	
0	X	5	8	11	9,4	11,5	X	8	→ $X_{n+3}(0)$
1	X	X	3	6	5,8	6,7	11		
2	X	3	X	3	5	4,3	8		
3	X	6	3	X	5,8	3	5		
4	X	5,8	5	5,8	X	3,5	9,4		
5	X	6,7	4,3	3	3,5	X	5,8		
6	X	X	X	X	X	X	X	0	← $Z_{n+3}(0)$;

Для пояснения центров притяжения маршрутов на рис. 2 показана аппроксимация траекторий маршрутов

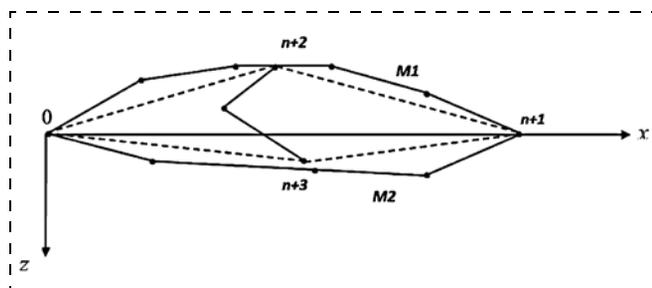


Рис. 2. Треугольная аппроксимация маршрутов M_1 и M_2

рутов M_1 и M_2 , состоящая из двух прямоугольных отрезков, исходящих из начала и конца маршрутов к этим центрам.

Назовем эту аппроксимацию треугольной моделью, элементы которой показаны штриховыми отрезками.

В данной работе на каждом k -м шаге расчета предлагается выполнять следующие действия, выделяя особо среди них новые:

- выбрать для анализа поочередно одну из двух матриц и найти в ней элемент минимальной длины, который нужно включить в план, как это принято в методе ветвей и границ;
- проанализировать найденный элемент (i, j) на "близость" к первому или второму маршруту;
- в выбранной одной из двух матриц удалить строку i и строку j , поставив знак запрета в элемент (j, i) , но дополнительно после присоединения элемента (i, j) уточнить координаты z и x центра притяжения соответствующего маршрута, поставив их в правый столбец выбранной матрицы;
- в другой матрице удалить две строки (i, j) и два столбца (i, j) для исключения их обслуживания.

Вследствие этих действий размерность обеих матриц снижается, а после их повторений число пересчетов каждой матрицы может оказаться неодинаковым, но сумма этих чисел в конце расчета будет равна $(n + 1)$. В итоге в предложенном подходе найдены ответы на два вопроса:

- в какой из двух матриц анализировать элементы малой длины;
- в какую из двух матриц ввести элемент минимальной длины.

Ответ таков — для анализа элементов по принципу равноправия обе матрицы используются поочередно, но затем элемент присоединяется к тому маршруту, к которому он "ближе", независимо от того, в какой матрице он был выбран.

На первом этапе расчета мера этой близости определяется разницей расстояний m_1 и m_2 от элемента (i, j) до центров притяжения $(n + 2)$ и $(n + 3)$ двух маршрутов, как показано на рис. 2.

Чтобы более детально понять сущность предлагаемых действий, рассмотрим решение показанного на рис. 1 примера при $n = 5$.

Пусть вначале центры притяжения двух маршрутов лежат посередине прямой линии $(0, n + 1)$, поскольку до распределения пунктов наблюдения их положение выше и ниже этой линии равновероятно.

На первом шаге ($k = 1$) выберем для анализа матрицу M_{10} , в ней минимальным элементом является дуга $(1, 2)$, которая также лежит на прямой $(0, 6)$ и одинаково близка к двум центрам притяжения. Поэтому возьмем в качестве решения первый маршрут ($p = 1$), т.е. в матрице M_{10} исключим строку 1 и столбец 2, поставив знак запрета в элемент $(2, 1)$, а в матрице M_{20} вычеркнем строки 1, 2 и столбцы 1, 2. Кроме того, в матрице M_{10} в дополнительном столбце $n + 1 = 7$ посчитаем новые ко-

ординаты центра притяжения $(n + 2)$, пользуясь следующими рассуждениями.

Так как дуга (i, j) имеет два конца, определим ее положение в системе координат x, z через координаты середины дуги, имеющие значения $(0,5x_i + 0,5x_j)$ и $(0,5z_i + 0,5z_j)$, что является апостериорной информацией о центре притяжения $(n + 2)$ маршрута M_1 по отношению к априорной, которой соответствуют значения $x_{n+2}(0) = 8$ и $z_{n+2}(0) = 0$, стоящие в правом столбце матрицы M_{10} . Поэтому искомый результат при $k = 1$ есть среднее значение между ними:

$$x_{n+2}(1) = 0,5x_{n+2}(0) + 0,25(x_i + x_j);$$

$$z_{n+2}(1) = 0,5z_{n+2}(0) + 0,25(z_i + z_j).$$

Можно записать в другом виде:

$$\begin{aligned} x_{n+2}(1) &= x_{n+2}(0) + \frac{0,5(x_i + x_j) - x_{n+2}(0)}{2}; \\ z_{n+2}(1) &= z_{n+2}(0) + \frac{0,5(z_i + z_j) - z_{n+2}(0)}{2}. \end{aligned} \quad (1)$$

Для имеющихся в матрице M_{10} числовых данных по формулам (1) получим $x_{n+2}(1) = 7,25$; $z_{n+2}(1) = 0$, и в итоге при $k = 1$ матрицы M_{11} и M_{21} имеют следующий вид:

▲ матрица M_{11}

	0	1	3	4	5	6	7
0	X	5	11	9,4	11,5	X	7,25
2	X	X	3	5	4,3	8	
3	X	6	X	5,8	3	5	
4	X	5,8	5,8	X	3,5	9,4	
5	X	6,7	3	3,5	X	5,8	
6	X	X	X	X	X	X	0

▲ матрица M_{21}

	0	3	4	5	6	7
0	X	11	9,4	11,5	X	8
3	X	X	5,8	3	5	
4	X	5,8	X	3,5	9,4	
5	X	3	3,5	X	5,8	
6	X	X	X	X	X	0

На втором шаге выбираем для анализа вторую матрицу вместо первой и в ней находим элемент $(5, 3)$ минимальной длины, который по физическому смыслу не должен попасть в оптимальные маршруты, но формально его нужно присоединить к одному из них, для чего необходимо установить, к какому из них он "ближе" и по какому критерию эту близость оценивать. На данном этапе расчета используем критерий минимума расстояний m_1 и m_2 от дуги (i, j) до каждого из двух центров притяжения $(n + 2)$ и $(n + 3)$, рассчитываемых по формулам:

$$\begin{aligned} m_1 &= \sqrt{(z_i - z_{n+2})^2 + (x_i - x_{n+2})^2}; \\ m_1 &= \sqrt{(z_i - z_{n+3})^2 + (x_i - x_{n+3})^2}. \end{aligned} \quad (2)$$

Для данного шага и элемента $(5, 3)$ оказывается, что $m_1 = 6$; $m_2 = 4$, что указывает на присоединение дуги к $(5, 3)$ к маршруту M_2 ($p = 2$). Поэтому, пересчитав по формуле (1) новые координаты центра притяжения второго маршрута и исключив строку 5 и столбец 3, а также элемент $(3, 5)$ в матрице M_{21} , а также удалив строки 3, 5 и столбцы 3, 5 в матрице M_{11} , получим две новые матрицы меньшей размерности:

▲ матрицу M_{12}

	0	1	4	6	7
0	X	5	9,4	X	7,25
2	X	X	5	8,5	
4	X	5,8	X	9,4	
6	X	X	X	X	0

▲ матрицу M_{22}

	0	4	5	6	7
0	X	9,4	11,5	X	9,5
3	X	5,8	X	5	
4	X	X	3,5	9,4	
6	X	X	X	X	0,75

На третьем шаге при очередном анализе матрицы M_{12} конкурентами на попадание в маршруты являются дуги $(0, 1)$ и $(2, 4)$, а из них минимальным элементом является дуга $(0, 1)$, и по критерию минимальных расстояний, вычисленных по формулам (2), она должна быть отнесена к маршруту M_1 .

Далее по принятой процедуре вначале надо исключить из матрицы M_{12} как столбец 1, так и строку 0, номер которой соответствует началу не только первого маршрута, но и второго. Поэтому затем из другой матрицы M_{22} строку 0 исключать нельзя, а нужно удалить только строку 1 и два столбца 0 и 1 (если они есть). В итоге получаем две новые матрицы:

▲ матрицу M_{13}

	0	4	6	7
2	X	5	8	5,7
4	X	X	9,4	
6	X	X	X	0

▲ матрицу M_{23}

	4	5	6	7
0	9,4	11,5	X	9,5
3	5,8	X	5	
4	X	3,5	9,4	
6	X	X	X	0,75

На четвертом шаге в результате чередования матриц анализируем матрицу M_{23} , в ней элементом минимальной длины является дуга $(4, 5)$, которая согласно формулам (2) ближе ко второму маршруту. В результате соответствующих действий получаем ответ в виде матриц M_{14} и M_{24} после исключения соответствующих строк и столбцов из матриц M_{13} и M_{23} :

▲ матрица M_{14}

	0	6	7
2	X	8	5,7
6	X	X	0

▲ матрица M_{24}

	4	6	7
0	9,4	X	9,5
3	5,8	5	
6	X	X	1,83

На пятом шаге при анализе матрицы M_{14} видим, что единственным элементом, претендующим на включение в оптимальный план, является дуга (2, 6), которую нужно отнести к первому маршруту. Поэтому после исключения из маршрута M_{24} только строки 6 (столбец 6 из матрицы не исключается, так как номер 6 соответствует концу не только первого, но и второго маршрутов) получаем матрицу M_{25} , а координаты центра притяжения первого маршрута равны $z_{n+2} = 0$, $x_{n+2} = 7,5$:

▲ матрица M_{25}

	4	6	7
0	9,4	X	9,5
3	5,8	5	1,83

В этой матрице элементом минимальной длины является дуга (3, 6), принадлежащая второму маршруту. Наконец, при исключении из матрицы M_{25} строки 3 и столбца 6 находим на шестом шаге последний для первого маршрута элемент (0, 4).

Объединяя полученные на всех шагах результаты, получаем, что маршруты M_1 и M_2 включают следующие пункты наблюдения:

$$M_1 \rightarrow 0, 1, 2, 6;$$

$$M_2 \rightarrow 0, 4, 5, 3, 6.$$

На рис. 3 эти маршруты представлены в виде траекторий, которые пока не соответствуют оптимальным из-за того, что в общий план попала дуга (5, 3).

На этом же рисунке представлено положение центров притяжения $(n+2) = 7$ и $(n+3) = 8$, которые удалось развести друг от друга в разные стороны при $k = 6$ с координатами:

$$x_{n+2}(6) = 7,25; x_{n+3}(6) = 9,2;$$

$$z_{n+2}(6) = 0; x_{n+3}(6) = 1,6.$$

Проведенный расчет позволяет сделать следующие промежуточные выводы:

1. Правило обязательного присоединения очередного элемента минимальной длины к одному из

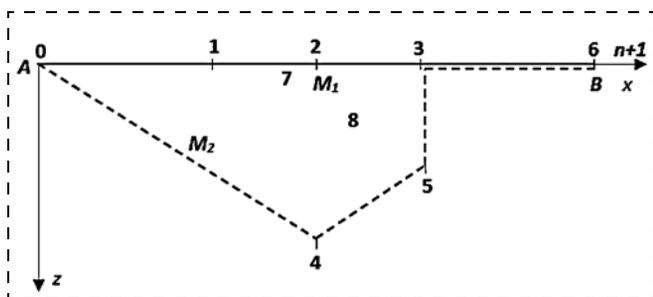


Рис. 3. Траектории маршрутов M_1 и M_2 , полученные на первом этапе расчета

маршрутов привело к тому, что в выбранные траектории попала дуга (5, 3), которая на самом деле является "перемычкой" между показанными на рис. 1 оптимальными маршрутами.

2. Сама перемычка (5, 3) обладает особыми свойствами — минимальные расстояния от нее до маршрутов M_1 и M_2 равны друг другу (от точки 3 до M_1 и от точки 5 до M_2) или, по крайней мере, мала их разность, а сама дуга оказалась между маршрутами. Другие элементы исходной матрицы этим свойством не обладают.

3. Достоинством предложенного выше расчета является способность развести маршруты M_1 и M_2 в разные стороны, как видно на рис. 3.

4. Недостатком оценки близости анализируемого элемента (i, j) к маршрутам с помощью расстояний m_1 и m_2 является то, что она полностью не учитывает взаимного расположения трех анализируемых объектов — M_1 , M_2 и (i, j) , а точнее, знаков минимальных разностей расстояний не до центров притяжения, а до самих траекторий треугольных моделей аппроксимации, как это показано на рис. 4 для четырех возможных ситуаций — $a, b, в, г$.

Видно, что в ситуации a не вызывает сомнения принадлежность дуги (i, j) маршруту M_2 , в ситуации b — маршруту M_1 , в то время как в ситуации $в$ и $г$ эти сомнения возникают, когда оба пункта i и j находятся либо вне маршрутов, либо между ними. Однако для вычисления новых расстояний l_1 и l_2 с учетом их знаков знания только исходной матрицы недостаточно. Нужно знать сами координаты местоположения пунктов, начала и конца маршрутов, а также центров притяжения, вычисленных на первом этапе расчета.

Нужно также отметить, что в данной работе дополнительная оценка близости расстояний l_1 и l_2 базируется пока что на аппроксимации маршрутов треугольными моделями, содержащими четыре прямоугольных отрезка 1, 2, 3, 4 (рис. 4).

Использование дополнительного критерия близости анализируемых участков полета к выбираемым маршрутам на втором этапе расчетов

Применим в качестве второй попытки дополнительные правила оценки близости дуги (i, j) к маршрутам M_1 и M_2 для принятия решений.

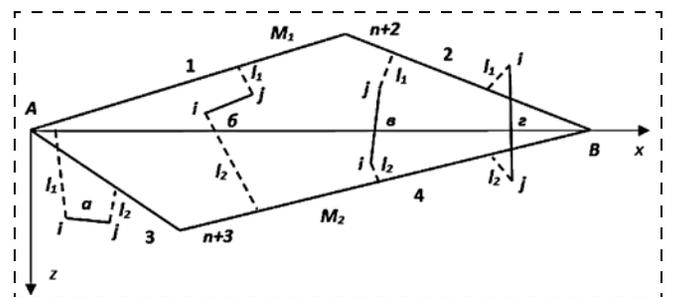


Рис. 4. Картина взаимного расположения дуги (i, j) и треугольных моделей аппроксимации маршрутов M_1 и M_2

1. Вычислить четыре разности для маршрута M_1 :

$$l_{11i} = z_i - z_{11}(x_i); l_{12i} = z_i - z_{12}(x_i);$$

$$l_{11j} = z_j - z_{11}(x_j); l_{12j} = z_j - z_{12}(x_j),$$

где z_{11} и z_{12} — координаты для прямоугольных отрезков 1 и 2, вычисленные по формулам

$$z_{11}(x_i) = x_i \frac{z_{n+2}}{x_{n+2}}; z_{12}(x_i) = \frac{z_{n+2}(x_i - x_{n+1})}{x_{n+2} - x_{n+1}};$$

$$z_{11}(x_j) = x_j \frac{z_{n+2}}{x_{n+2}}; z_{12}(x_j) = \frac{z_{n+2}(x_j - x_{n+1})}{x_{n+2} - x_{n+1}}. \quad (3)$$

2. Найти одну минимальную по модулю разность Δ_1 из четырех вычисленных ($l_{11i}, l_{11j}, l_{12i}, l_{12j}$) и запомнить ее с учетом знака.

3. Вычислить другие четыре разности для маршрута M_2 .

$$l_{23i} = z_i - z_{23}(x_i); l_{24i} = z_i - z_{24}(x_i);$$

$$l_{23j} = z_j - z_{23}(x_j); l_{24j} = z_j - z_{24}(x_j),$$

где z_{23} и z_{24} — координаты для прямоугольных отрезков 3 и 4, вычисленные по формулам

$$z_{23}(x_i) = x_i \frac{z_{n+3}}{x_{n+3}}; z_{24}(x_i) = \frac{z_{n+3}(x_i - x_{n+1})}{x_{n+3} - x_{n+1}};$$

$$z_{23}(x_j) = x_j \frac{z_{n+3}}{x_{n+3}}; z_{24}(x_j) = \frac{z_{n+3}(x_j - x_{n+1})}{x_{n+3} - x_{n+1}}. \quad (4)$$

4. Найти одну минимальную по модулю разность Δ_2 из четырех вычисленных на предыдущем шаге, и с учетом ее знака сравнить с Δ_1 , присваивая следующие значения вспомогательной переменной p :

- если Δ_1 и Δ_2 имеют одинаковые знаки, то при $|\Delta_1| \leq |\Delta_2|$: $p = 1$ (присоединить (i, j) к маршруту M_1); при $|\Delta_1| > |\Delta_2|$: $p = 2$ (присоединить (i, j) к маршруту M_2);
- если Δ_1 и Δ_2 имеют разные знаки, то при $|\Delta_1 + \Delta_2| \geq \delta$ и $|\Delta_1| \leq \Delta_2$: $p = 1$;
- при $|\Delta_1 + \Delta_2| \geq \delta$ и $|\Delta_1| > \Delta_2$: $p = 2$;

$$\text{при } |\Delta_1 + \Delta_2| < \delta: p = 0$$

(ни одному маршруту дуга (i, j) не принадлежит, поставить в элементы (i, j) (j, i) обеих матриц M_1 и M_2 знаки запрета). Величина δ назначается экспериментальным путем, например, пусть $\delta = 0,1|z_{n+2} - z_{n+3}|$.

Тогда решение первого примера во втором этапе расчета выглядит так. Учитывая результаты первого этапа, показанные на рис. 3, повторим действия по шагам.

На первом шаге очевидно, что дуга (1, 2) принадлежит маршруту M_1 , так как $\Delta_1 = 0$, $\Delta_2 = 3,1$. Зато на втором шаге при анализе во второй матрице элемента (5, 3) получим $\Delta_1 + \Delta_2 = 0,3$, значит $p = 0$, и согласно правилу (5) эта перемычка исключается из маршрутов. Поэтому после запрещения элемен-

тов (5, 3) и (3, 5) матрицы M_{12} и M_{22} приобретают следующий вид:

▲ матрица M_{12}

	0	1	3	4	5	6	7
0	X	5	11	9,4	11,5	X	7,25
2	X	X	3	5	4,3	8	
3	X	6	X	5,8	X	5	
4	X	5,8	5,8	X	3,5	9,4	
5	X	6,7	X	3,5	X	5,8	
6	X	X	X	X	X	X	0

▲ матрица M_{22}

	0	3	4	5	6	7
0	X	11	9,4	11,5	X	8
3	X	X	5,8	X	5	
4	X	5,8	X	3,5	9,4	
5	X	X	3,5	X	5,8	
6	X	X	X	X	X	0

На третьем шаге планирование осуществляется по новому пути. Так как на предыдущем шаге окончательное решение $p = 1, 2$ не было найдено, дадим еще раз возможность выбора матрице M_{22} , при анализе которой элементом минимальной длины оказалась дуга (4, 5), которую нужно отнести к маршруту M_2 . Поэтому, исключив из матрицы M_{22} элемент (5, 4), строку 4 и столбец 5, а из матрицы M_{12} — строки 4, 5 и столбцы 4, 5, получим:

▲ матрица M_{13}

	0	1	3	6	7
0	X	5	11	X	7,25
2	X	X	3	8	
3	X	6	X	5	
6	X	X	X	X	0

▲ матрица M_{23}

	0	3	4	6	7
0	X	11	9,4	X	8,75
3	X	X	5,8	5	
5	X	X	X	5,8	
6	X	X	X	X	2

На четвертом шаге переходим к анализу матрицы M_{13} , дающей ответ — присоединить дугу (2, 3) к маршруту M_1 , исключив из матрицы M_{13} строку 2 и столбец 3, а из матрицы M_{23} — строку 3 и столбец 3, после чего получим:

▲ матрица M_{14}

	0	1	6	7
0	X	5	X	9
3	X	6	5	
6	X	X	X	1,3

▲ матрица M_{24}

	0	4	6	7
0	X	9,4	X	8,75
5	X	X	5,8	
6	X	X	X	2

На пятом шаге при анализе матрицы M_{24} находим элемент (5, 6), принадлежащий маршруту M_2 , а на шестом шаге, анализируя матрицу M_{14} , выбираем элемент (0, 1), присоединяя его к маршруту M_1 . В конце решения задачи однозначно определяются

элементы (0, 4) для маршрута M_2 и (3, 6) — для маршрута M_1 , что приводит к желаемому оптимальному результату, показанному на рис. 1.

Пример координированного планирования двух маршрутов, находящихся близко друг от друга

Чтобы ярче показать эффект от избавления перемычек и использования треугольной модели траекторий маршрутов, рассмотрим второй пример, которому соответствует рис. 5.

Необходимо обслужить двумя ЛА 10 пунктов, между которыми самыми короткими являются перемычки (1, 2), (3, 4), (5, 6), (7, 8), (9, 10) длиной $r = 3$, диагональные дуги (2, 3), (4, 5), (6, 7), (8, 9) имеют длину $r = 5$, а две дуги (9, 10), (10, 11) имеют длину $r = 6$.

При решении этой задачи известным методом одномерной маршрутизации выясняется, что перемычки длиной $r = 3$ являются лидерами в отборе элементов минимальной длины, и они должны попасть в маршрут. Тогда после исключения соответствующих строк и столбцов из исходной матрицы размерности 12×12 можно убедиться, что в сформированной матрице размерности 7×7 строки имеют четные номера, столбцы — нечетные, а следующими элементами минимальной длины становятся диагональные дуги.

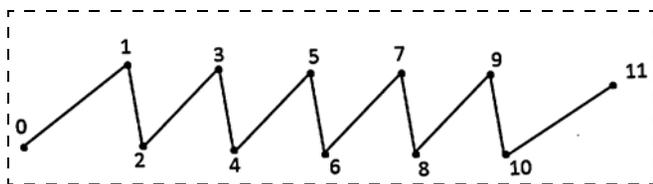


Рис. 5. Пример траектории полета, полученной при одномерной маршрутизации

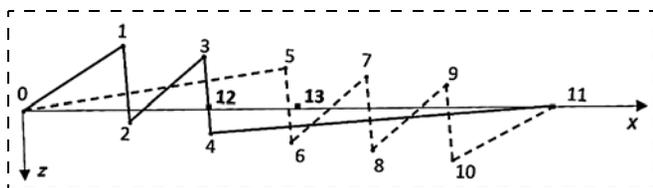


Рис. 6. Траектория двух маршрутов полета на первом этапе расчетов при $p = 1, 2$

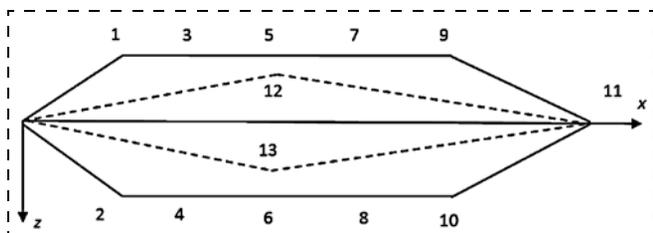


Рис. 7. Траектории двух маршрутов полета на втором этапе расчета при $p = 0, 1, 2$

После их включения в маршрут становится ясно, что остались неназванными дуги (0, 1) и (10, 11). Вид найденной траектории на рис. 5 указывает, что полученное решение далеко от идеала.

При решении задачи предложенным методом двумерной маршрутизации на первом этапе расчета формируются две исходных матрицы расстояний для двух ЛА размерностью 12×12 . Не представляя их для краткости изложения, можно убедиться, что при их поочередном анализе и двухальтернативном выборе номера маршрута ($p = 1, 2$) для каждой дуги (i, j) полученный ответ также далек от идеала. На рис. 6 непрерывной линией показан первый маршрут M_1 , штриховой линией — второй маршрут.

Также оказалось, что центры притяжения $n + 2 = 12$ и $n + 3 = 13$ разнесены друг от друга и имеют координаты

$$x_{12} = 9,5; z_{12} = 0; x_{13} = 15,3; z_{13} = 0.$$

Результаты анализа на втором этапе расчета вносят существенные изменения в процесс маршрутизации за счет трехальтернативного выбора ($p = 0, 1, 2$), а именно — вначале на первых пяти шагах расчета все перемычки длиной $r = 3$ согласно второму, дополнительному критерию близости не попадают в план полета, в силу чего конкурентами становятся элементы (1, 3), (3, 5), (5, 7), (7, 9), (2, 4), (4, 6), (6, 8), (8, 9), которые имеют длину $r = 4$ и при известном способе даже не попадали в расчеты.

Далее на последующих восьми шагах поочередного анализа двух матриц первые четыре из перечисленных выше элементов присваиваются первому маршруту, а остальные — второму маршруту, в результате чего формируются две матрицы:

▲ матрица $M_{1, 14}$

	1	2	11	
0	6	6	X	10,7
9	16	16,2	6	
10	16,2	16	6	-1,2

▲ матрица $M_{2, 14}$

	1	2	11	
0	6	6	X	10,7
9	16	16,2	6	
10	16,2	16	6	1,2

Неназванными для анализа остались элементы этих матриц, в том числе на последних трех шагах расчета автоматически выбираются элемент (0, 1) для первого маршрута и элемент (10, 11) для второго маршрута. Сформированный план двумерной маршрутизации готов и показан на рис. 7.

На рис. 7 также показано окончательное расположение центров притяжения 12 и 13 принятой в данной работе треугольной модели аппроксимации траекторий полета, показанных штриховыми линиями:

$$x_{12} = 13,7; z_{12} = -1,07; x_{13} = 14,6; z_{13} = 1,07.$$

Единый перечень действий алгоритма двумерной маршрутизации

Перечислим основные действия алгоритма на двух этапах расчета, выделив новые из них подчеркиванием.

На первом этапе:

1. Из исходной матрицы M_0 расстояний между пунктами с установленными в нужных местах запрещенными элементами сформировать две матрицы M_{10} и M_{20} , в каждой из которых в дополнительном правом столбце указать одинаковые по величине начальные координаты центров притяжения — в матрице M_{10} координаты:

$$x_{n+2}(0) = 0,5r(0, n+1), z_{n+2}(0) = 0;$$

в матрице M_{20} координаты:

$$x_{n+3}(0) = 0,5r(0, n+1), z_{n+3}(0) = 0.$$

2. Выполнить поочередно анализ одной из матриц известным в методе ветвей и границ способом:

- выявить элементы минимальной длины в каждой строке и в каждом столбце;
- выбрать из них тот элемент (i, j) , исключение которого из маршрута приведет к наибольшему штрафу.

3. В отличие от известного метода решить вопрос — в какой из двух маршрутов нужно включить элемент (i, j) , не взирая на то, какая матрица анализировалась. Использовать для этого первый критерий близости дуги (i, j) к центрам притяжения $(n+2)$ и $(n+3)$ двух маршрутов путем вычисления расстояний m_1 и m_2 по формулам (2). Далее применить двухальтернативное правило:

если $m_1 - m_2 \leq 0$ — включить (i, j) в первый маршрут ($p = 1$),

если $m_1 - m_2 > 0$ — включить (i, j) во второй маршрут ($p = 2$).

4. В матрице того маршрута, куда попал элемент (i, j) , вычеркнуть строку i , столбец j и элемент (j, i) . В другой матрице вычеркнуть две строки i и j и два столбца i и j , за исключением случаев, указанных в п. 5.

5. Если минимальными оказались элементы $(0, j)$ или элемент $(i, n+1)$, т.е. в них участвуют общие начальные или конечные маршруты, то в первый раз соответствующие строки и столбцы в другой матрице не исключаются, а эти действия разрешаются, если элементы попадают в число минимальных дважды.

6. После включения нового элемента в выбранную матрицу в ее правом столбце указать новые координаты z и x соответствующего центра притяжения, вычисленные по формулам (1).

7. В результате перечисленных действий размерность обеих матриц уменьшается, и процесс расчета повторяется при переходе в п. 2 до тех пор, пока после $(n+1)$ шагов матрицы не превратятся в один элемент, а координаты центров притяжения не будут удалены друг от друга на некоторое расстояние, достаточное для второго этапа вычислений.

На втором этапе:

1. Вновь повторить формирование исходных матриц M_{10} и M_{20} , указав, однако, в дополнительном правом столбце каждой из них неодинаковые координаты центров притяжения, полученные на первом этапе расчета.

2. Выполнить поочередно анализ одной из матриц известным способом так же, как и на первом этапе.

3. В отличие от первого этапа решить задачу трехальтернативного выбора ($p = 0, 1, 2$), используя второй дополнительный критерий близости элемента (i, j) к самим траекториям маршрутов, описываемым треугольной моделью, при вычислении минимальных расстояний Δ_1 и Δ_2 по формулам (3)—(4). Далее использовать трехальтернативное правило (5):

если $\text{sign}\Delta_1 = \text{sign}\Delta_2$ и $|\Delta_1| \leq |\Delta_2|$ — включить (i, j) в первый маршрут ($p = 1$);

если $\text{sign}\Delta_1 = \text{sign}\Delta_2$ и $|\Delta_1| > |\Delta_2|$ — включить (i, j) во второй маршрут ($p = 2$);

если $\text{sign}\Delta_1 \neq \text{sign}\Delta_2$; $|\Delta_1 + \Delta_2| > \delta$ и $\Delta_1 \leq \Delta_2$ — включить (i, j) в первый маршрут ($p = 1$);

если $\text{sign}\Delta_1 \neq \text{sign}\Delta_2$; $|\Delta_1 + \Delta_2| > \delta$ и $\Delta_1 > \Delta_2$ — включить (i, j) во второй маршрут ($p = 2$);

если $\text{sign}\Delta_1 \neq \text{sign}\Delta_2$; $|\Delta_1 + \Delta_2| < \delta$ — исключить попадание (i, j) в оба маршрута ($p = 0$).

4. Если $p = 1, 2$, действия по исключению строк и столбцов из двух матриц аналогичны п. 4 первого этапа расчетов.

5. Если $p = 0$, то в двух матрицах необходимо исключить элемент (i, j) и (j, i) , и проанализированной матрице снова предоставляется право выбрать другой элемент минимальной длины.

6. По включению нового элемента в один из маршрутов в правом столбце соответствующей матрицы указать новые координаты центра притяжения (аналогично п. 6 первого этапа расчета).

7. В результате перечисленных действий процесс расчета повторяется при переходе в п. 2 до тех пор, пока не будут найдены оба окончательных маршрута.

Обобщение предложенного подхода на случаи многомерной маршрутизации полета группы ЛА

Предложенный перечень действий двухэтапного расчета может быть распространен для задачи координированного выбора большего, чем два, числа N маршрутов полета. Тогда из исходной матрицы расстояний тиражируется N матриц M_{p0} ($p = 1, \dots, N$), которые анализируются поочередно.

Тогда на первом этапе расчета на каждом его шаге выбирается один элемент (i, j) минимальной длины, который затем присваивается одному из маршрутов по первому критерию близости, в силу чего вычисляются новые координаты его центра притяжения, а в остальных матрицах строки i, j и столбцы i, j исключаются. По мере неизбежного уменьшения размерности анализируемых матриц центры притяжения траекторий полета станут неоди-

наковыми, что достаточно для продолжения решения задачи.

Выводы

1. Предложен способ решения задачи двумерной маршрутизации, предусматривающий формирование из исходной матрицы расстояний между пунктами двух матриц для каждого разомкнутого маршрута, которые анализируются поочередно при координированном выборе элемента минимальной длины в каждой из них.

При этом одновременно из другой матрицы исключаются две соответствующие строки и два столбца, чтобы избежать повторного обслуживания пунктов.

2. Наряду с выбором нужного элемента в решение введена новая операция выбора того маршрута, который должен этот элемент присоединить, с использованием при этом двух критериев близости элемента к каждому из маршрутов:

- близость к центру притяжения треугольной модели аппроксимации траектории полета;
- близость к самой траектории при оценке минимального расстояния до него.

3. Расчет состоит из двух этапов — первый этап не учитывает возможную близость двух траекторий и нежелательное включение в план коротких перемычек между ними. Целью этого этапа является разведение центров притяжения маршрутов в разные стороны.

Целью второго этапа является оценка положения выбранного элемента матрицы относительно аппроксимированных траекторий и, в отличие от известного подхода, отказ в ряде случаев от его включения в любой из маршрутов во избежание их лишних пересечений.

4. Предложено обобщение рассмотренного подхода на многомерную маршрутизацию и на случай планирования замкнутых маршрутов, что позволило оценить выигрыш во времени выполнения полета для разомкнутых маршрутов в 2,5...3 раза, для замкнутых маршрутов — в 1,5...2 раза.

5. Как и метод ветвей и границ, предложенный способ расчета является весьма трудоемким. По-

этому для его реализации на борту ЛА в реальном времени необходимо использовать нейросетевые технологии на этапе их обучения.

Список литературы

1. Лебедев Г. Н., Мирзоян Л. А., Ефимов А. В. Нейросетевое планирование групповых действий ЛА при наблюдении заданной группы подвижных объектов // Мехатроника, автоматизация, управление. 2009. № 11. С. 60—65.
2. Лебедев Г. Н., Ефимов А. В. Применение динамического программирования для маршрутизации облета подвижных объектов в контролируемом регионе // Труды СГАУ. 2012. № 1. С. 63—70.
3. Лебедев Г. Н., Мирзоян Л. А. Маршрутизация полета БЛА с учетом его динамики при наблюдении неподвижных наземных объектов // Мехатроника, автоматизация, управление. 2011. № 12. С. 24—28.
4. Кузин Л. Т. Основы кибернетики. М.: Энергия Т. 1 и Т. 2, 1973.
5. Кюнчи Г. Н., Крилле В. Нелинейное программирование. М.: Сов. радио, 1965.
6. Гришанин Ю. С., Лебедев Г. Н., Липатов А. В., Степаныч Г. А. Теория оптимальных систем. М.: МАИ, 1999. 317 с.
7. Лебедев Г. Н., Тин Пхон Чжо, Зо Мин Тайк, Хахулин Г. Ф., Малыгин В. Б. Оптимальное управление и контроль безопасности поперечного движения речных и воздушных судов при пересечении их маршрутов // Мехатроника, автоматизация, управление. 2012. № 12. С. 50—55.
8. Лебедев Г. Н., Зо Мин Тайк. Синтез оптимального управления боковым движением воздушных или речных судов при пересечении их маршрутов под произвольным углом. М.: Новые технологии, 2014. № 5. С. 61—68.
9. Лебедев Г. Н., Румакина А. В. Система логического управления обхода препятствий беспилотным летательным аппаратом при маршрутном полете // Труды МАИ. 2015. Вып. 83. С. 1—19.
10. Беллман Р. Динамическое программирование. М.: ИИЛ, 1961. 400 с.
11. Лебедев Г. Н., Мирзоян Л. А. Нейросетевое планирование действий по облету наземных объектов группой летательных аппаратов // Авиакосмическое приборостроение. 2005. № 12.
12. Лебедев Г. Н. Методы принятия оперативных решений в задачах управления и контроля. М.: МАИ, 1992.
13. Летов А. М. Динамика полета и управление. М.: Наука, 1969. 360 с.
14. Гасс С. Линейное программирование. М.: Физматгиз, 1961. 304 с.
15. Понтрягин Л. С. Принцип максимума в оптимальном управлении. М.: Наука, 1989. 62 с.
16. Понтрягин Л. С. Математическая теория оптимальных процессов. М.: Наука, 1983. 393 с.

Modification of the Integer Branch and Bound Method for Solving of a Two-Dimensional Routing Task of an Aircraft Group's Coordinated Flight

G. N. Lebedev, D. Sc., Professor, e-mail: kaf301@mai.ru,

V. I. Goncharenko, D. Sc., Associate Professor, A. V. Roumakina, Assistant
Moscow Aviation Institute (National Research University), Moscow, 125993, Russian Federation

Corresponding author: Roumakina Alena V., Assistant,
Moscow Aviation Institute (National Research University), Moscow, 125993, Russian Federation
E-mail: kaf301@mai.ru

Below is a solution for a direct-sales representative in case of a two-dimensional routing of a flight, which differs by formation of two initial matrixes of the distances between the points of two unlinked routes, which are analyzed in turn with a coordinated selection of the element of the minimal length in each of them. In comparison with the known branch and bound method, there is no need in an additional choice for an alternative, which is essentially new in the solved task — to which of the two routes the element should belong to. For this purpose it is necessary to know, to what route it is closer to, and for evaluation of this proximity two criteria are used: — at the first stage of calculation on each step it is the calculation of the minimal distance from an element to the center of gravity of each of the routes by using the triangular model of approximation of the flight trajectory. This allows us to part gradually step by step the centers of gravity at a certain distance, sufficient to continue the calculations; — at the second stage of calculation, the minimal distances from the element to the approximated trajectories of each route are calculated, which allows us to refuse in case of their equality in general accession of the element to any route and by that to avoid an unnecessary inclusion of the crossing points between them. Examples of the calculations proving the efficiency of the offered approach and indicating reduction of the performance of the flight of points more than twice are presented.

Keywords: unmanned flying vehicle, multi-dimensional routing, in-flight safety, integer programming

Acknowledgements: This work was supported by RFBR material 16-08-00-832-a, 15-08-00043 grants

For citation:

Lebedev G. N., Goncharenko V. I., Roumakina A. V. Modification of the Integer Branch and Bound Method for Solving of a Two-Dimensional Routing Task of an Aircraft Group's Coordinated Flight, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2016, vol. 17, no. 11, pp. 783—791.

DOI: 10.17587/mau.17.783-791

References

1. **Lebedev G. N., Mirzozjan L. A., Efimov A. V.** *Nejroseteivoje planirovanie gruppovyh dejstvij LA pri nablyudenii zadannoj gruppy podvizhnyh obektov* [Neural network planning group actions of aircraft under the supervision of a given group of mobile objects], *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2009, no.11, pp. 60—61.
2. **Lebedev G. N., Efimov A. V.** *Primenenie dinamicheskogo programmirovaniya dlya marshrutizatsii obleta podvizhnyh obektov v kontroliruemom regione* [Use dynamic programming to route over moving objects in the monitored region], *Trudy SGAU*, 2012, no. 1.
3. **Lebedev G. N., Mirzoyan L. A.** *Marshrutizatsiya poleta BLA s uchetom ego dinamiki pri nablyudenii nepodvizhnyh nazemnyh obektov* [Routing of the UAV taking into account its dynamics when the observation of fixed objects], *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2011, no. 12, pp. 24—28.
4. **Kuzin L. T.** *Osnovy kibernetiki* [Foundations of Cybernetics], Moscow, Ehnergiya, 1973.
5. **Kyunci G. N., Krille V.** *Nelinejnoe programmirovanie* [Nonlinear programming]. Moscow: Publishing house "Sov. radio", 1965.
6. **Grishanin Ju.S., Lebedev G. N., Lipatov A. V., Stepan'janc G. A.** *Teoriya optimal'nyh sistem* [The theory of optimal systems], Moscow, Publishing house "MAI", 1999, 317 p.

7. **Lebedev G. N., Tin Phon Chzho, Zo Min Tajk, Hahulin G. F., Malygin V. B.** *Optimal'noe upravlenie i kontrol' bezopasnosti poperechnogo dvizheniya rechnyh i vozdushnyh sudov pri peresechenii ih marshrutov* [Optimal control and security control lateral movement of river vessels and aircraft crossing their routes], *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2012, no. 12, pp. 50—55.

8. **Lebedev G. N., Zo Min Tajk.** *Sintez optimal'nogo upravleniya bokovym dvizheniem vozdushnyh ili rechnyh sudov pri peresechenii ih marshrutov pod proizvol'nym uglom* [Synthesis of optimal control of the lateral movement of air or sea vessels crossing their routes at any angle], *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2014, no. 5, pp. 61—68.

9. **Lebedev G. N., Rumakina A. V.** *Sistema logicheskogo upravleniya obhoda prepyatstvij bespilotnym letatel'nom apparatom pri marshrutnom polete* [Logic control system to avoid obstructions unmanned aerial vehicle during cross-country flights], *Trudy MAI*, 2015, iss. 83.

10. **Bellman R.** *Dinamicheskoe programmirovanie* [Dynamic programming], Moscow, Foreign literature, 1961, 400 p.

11. **Lebedev G. N., Mirzoyan L. A.** *Neyroseteivoje planirovanie deystviy po obletu nazemnykh obektov gruppy letatel'nykh apparatov* [Neuronet action planning for flyby terrestrial objects, by a group of aircraft], *Aviakosmicheskoe priborostroenie*, 2005, no. 12.

12. **Lebedev G. N.** *Metody prinyatiya operativnykh resheniy v zadachakh upravleniya i kontrolya* [Methods for operational solution of the problems of management and control], Moscow: Publishing house "MAI", 1992.

13. **Letov A. M.** *Dinamika poleta i upravleniya* [Flight dynamics and control], Moscow, Nauka, 1969, 360 p.

14. **Gass S.** *Lineynoe programmirovanie* [Linear programming], Moscow, Fizmatgiz, 1961, 304 p.

15. **Pontryagin L. S.** *Printsip maksimuma v optimal'nom upravlenii* [The maximum principle in optimal control], Moscow, Nauka, 1989, 62 p.

16. **Pontryagin L. S.** *Matematicheskaya teoriya optimal'nykh protsessov* [Mathematical Theory of Optimal Processes], Moscow, Nauka, 1983, 393 p.



29—30 июня 2017г., Казахстан, г. Астана
**МЕЖДУНАРОДНАЯ IEEE-ЕВРАЗИЙСКАЯ
КОНФЕРЕНЦИЯ ПО ЭНЕРГЕТИКЕ,**
приуроченная к международной выставке
ASTANA EXPO-2017



**И МЕЖДУНАРОДНАЯ IEEE-СИБИРСКАЯ
КОНФЕРЕНЦИЯ ПО УПРАВЛЕНИЮ И СВЯЗИ**

Международная IEEE-Евразийская конференция по энергетике, приуроченная к международной выставке ASTANA EXPO-2017, и тринадцатая IEEE-Сибирская конференция, посвященная достижениям в области разработки и создания систем управления и связи, регулярно организуются Красноярской и Томской группой и студенческим отделением IEEE, компанией National Instruments для того, чтобы поддерживать междисциплинарные дискуссии и взаимодействие среди ученых и инженеров, развивать международное сотрудничество через участие в деятельности профессиональных сообществ Института IEEE.

Конференция будет проходить в Казахском агротехническом университете им. Сакена Сейфуллина (КАТУ), являющимся старейшим вузом в Астане.

Это современный, динамично развивающийся университет, один из лидеров в системе высшего образования Казахстана.

*Председатель конференции — ректор КАТУ, профессор **А. К. Куришбаев***

НАПРАВЛЕНИЯ РАБОТЫ КОНФЕРЕНЦИИ

1. Фундаментальные проблемы теории управления и связи.
2. Энергосбережение и энергетика будущего.
3. Компьютерные измерительные технологии, датчики и системы.

Программа конференции предусматривает заседания секций с устными докладами, специальные заседания, краткие курсы и культурную программу.

Международная специализированная выставка Astana EXPO-2017 будет посвящена энергетике будущего (www.expo2017astana.com).

Компания National Instruments организует выставку и проводит симпозиум "Компьютерные измерительные технологии", целью которого является обсуждение вопросов внедрения инновационных технологий для решения инженерных задач и их использования для автоматизации производства, моделирования, обработки результатов научного эксперимента.

*Подробную информацию о конференции см. на сайте:
sibcon.sfu-kras.ru, www.kazatu.kz, ieee.tpu.ru/sibcon*

Издательство «НОВЫЕ ТЕХНОЛОГИИ»

107076, Москва, Стромынский пер., 4

Телефон редакции журнала: **(499) 269-5397**, тел./факс: **(499) 269-5510**

Технический редактор *Е. В. Конова*. Корректор *Е. В. Комиссарова*.

Сдано в набор 05.09.2016. Подписано в печать 12.10.2016. Формат 60×88 1/8. Бумага офсетная.
Усл. печ. л. 8,86. Заказ МН1116. Цена договорная.

Журнал зарегистрирован в Комитете Российской Федерации по делам печати, телерадиовещания и средств массовых коммуникаций
Свидетельство о регистрации ПИ № 77-11648 от 21.01.02

Учредитель: Издательство "Новые технологии"

Оригинал-макет ООО "Авансед солюшнз". Отпечатано в ООО "Авансед солюшнз".
119071, г. Москва, Ленинский пр-т, д. 19, стр. 1.