Научно-технический журнал
Выходит 4 раза в год

УЧРЕДИТЕЛЬ

Институт проблем управления им. В.А. Трапезникова РАН

Главный редактор
И.В. Прангишвили

Заместители главного редактора
А.Н. Шубин, Ф.Ф. Пащенко

Ответственный секретарь
Л.П. Боровских

Региональные редсоветы (руководители)

Владивосток — О.В. Абрамов (4232) 31-02-02

Липецк — Л.А. Кузнецов (0742) 32-80-44

Издатель ООО «СенСиДат»

Ген. директор Н.Н. Кузнецова

Адрес редакции
117997, ГСП-7, Москва,
ул. Профсоюзная, д. 65, к. 104.
Тел./факс (095) 330-42-66,
тел.: (095) 334-92-00,
334-90-20

E-mail: datchik@ipu.ru
www.ipu.ru/period/pu

Оригинал-макет
и электронная версия
подготовлены ООО «АКИМ»

Отпечатано с готовых
диапозитивов в ООО «Риконта»,
г. Москва, Волгоградский пр-т,
д. 26, стр. 1

Подписано в печать
26.07.2004 г.

Заказ № 642

Журнал зарегистрирован
в Министерстве
Российской Федерации
по делам печати,
телерадиовещания
и средств массовых
коммуникаций

Свидетельство о регистрации
ПИ №77-11963
от 06 марта 2002 г.

Подписные индексы:
81708 в каталоге Роспечати
38006 в объединенном каталоге
«Пресса России»

© СенСиДат, 2004 г.

ПРОБЛЕМЫ УПРАВЛЕНИЯ

3.2004

СОДЕРЖАНИЕ

Обзоры

Кудинов Ю. И., Дорохов И. Н., Пащенко Ф. Ф. Нечеткие регуляторы и системы управления 2

Системный анализ и обработка данных

Гребенюк Е. А. Мониторинг нестационарных процессов: анализ и исследование изменений свойств стационарности 15
Рыков А. С., Исходжанов Р. Р. Многокритериальная оптимизация портфеля инвестиций. Детерминированный случай. 21

Управление в социально-экономических системах

Кононов Д. А., Кульба В. В., Шубин А. Н. Информационное управление в социально-экономических системах: элементы управления и способы информационного воздействия 25
Агеев И. А., Гурлев И. В., Цыганов В. В. Механизмы манипулирования корпорацией. 34

Управление технологическими процессами

Ивашкин Ю. А. Структурно-параметрическое моделирование и идентификация аномальных ситуаций в сложных технологических системах. 39
Лебедев В. Г. Принципы построения интеллектуального интерфейса пользователя для систем поддержки принятия решений оператором. 43

Надежность и техническая диагностика

Ведешенков В. А. Самодиагностирование возникающих отказов и их устранение в цифровых системах с реконфигурацией 48

Управление подвижными объектами

Микрин Е. А., Суханов Н. А., Платонов В. Н. и др. Принципы построения бортовых комплексов управления автоматических космических аппаратов. 62

Информационные технологии в управлении

Выхованец В. С., Малюгин В. Д. Мультипликативная алгебра и ее применение в логической обработке данных 67
Стецюра Г. Г. Работа локальных комплексов GRID в режиме жесткого реального времени 78

Хроника

Станислав Васильевич Емельянов (к 75-летию со дня рождения) . . 83
Евгений Александрович Федосов (к 75-летию со дня рождения) . . . 86

УДК 658.012.011.56

НЕЧЕТКИЕ РЕГУЛЯТОРЫ И СИСТЕМЫ УПРАВЛЕНИЯ[1]

Ю. И. Кудинов*, И. Н. Дорохов**, Ф. Ф. Пащенко***

*Липецкий государственный технический университет;
**Российский химико-технологический университет им. Д. И. Менделеева, г. Москва;
***Институт проблем управления им. В. А. Трапезникова, г. Москва

Рассмотрены вопросы анализа, синтеза и классификации нечетких регуляторов и систем управления. Дан анализ публикаций по нечетким регуляторам. Основное внимание уделено логико-лингвистическим, аналитическим и обучаемым нечетким регуляторам и системам управления. Рассмотрена нечеткая разностная TS-модель, пригодная для описания динамических объектов и нечетких регуляторов. Показано применение методов Ляпунова для оценки устойчивости нечетких систем и синтеза нечетких регуляторов. Приведены структура и методы обучения нечетких систем управления, содержащих нейронечеткие TS-модели регулятора и объекта. Дано сравнение нечетких и статистических регуляторов.

ВВЕДЕНИЕ

В теории управления особое внимание всегда уделялось проблеме синтеза математических моделей и алгоритмов управления при недостаточной информации об объекте управления и действующих на него полезных сигналов и помех. Этот интерес усилился в последнее время в связи с изучением слабо формализованных сложных систем и разработкой принципов и алгоритмов управления этими системами.

Опыт создания систем автоматического управления для сложных технологических объектов, в условиях большой неопределенности и неполноты знаний об объекте, нечеткости описаний показал неэффективность применения только формальных классических методов теории управления. Этим объясняется повышенный интерес к невероятностным моделям нечеткости и неопределенности, характерный для 1960—1970-х гг. Субъективная вероятность Севеджа, верхние и нижние вероятности Демпстера, характеризующие неопределенность в теории вероятностей, правдоподобие и доверие Шеффера и, наконец, теория нечетких множеств Заде — это неполный список направлений, ориентированных на моделирование на основе экспертных знаний и принятие решений в условиях нечеткости и неопределенности.

В последнее время наблюдается исключительно высокий интерес к одному из важнейших приложений теории нечетких множеств — анализу и синтезу нечетких регуляторов и систем управления технологическими процессами и установками. Нужно сказать, что широко применяемые традиционные ПИ- и ПИД-алгоритмы можно считать экспертными, так как были впервые построены в конце XIX столетия, когда теории автоматического управления еще не существовало. Отечественные публикации по этой проблематике относятся к концу 1980-х и началу 1990-х гг. [1—7]. Вместе с тем, за прошедшее десятилетие произошло лавинообразное увеличение числа теоретических и прикладных исследований в области нечетких регуляторов и систем управления. Об этом свидетельствуют последние конгрессы IFAC и появление новых журналов, посвященных этой проблеме.

Большое число практических применений нечетких систем управления в промышленности, транспорте и бытовых приборах отмечено в Японии, Китае, США и в европейских странах.

В последнее время уже более определенно можно говорить о сформировавшихся трех видах нечетких регуляторов: логико-лингвистических, аналитических и обучаемых. Однако информация о

[1] Работа выполнена при поддержке РФФИ по проекту 04-01-00816 и ОЭММПУ РАН по программе № 16.

них не систематизирована и разбросана по многим публикациям. Цель настоящей работы — дать конструктивный анализ нечетких регуляторов, позволяющий специалисту свободно ориентироваться в области нечетких систем управления технологическими процессами в условиях неопределенности.

ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ

Рассмотрим такие основные понятия, как нечеткие множества и некоторые операции на них, лингвистическая переменная и нечеткое отношение.

Нечетким множеством X на универсальном множестве $\mathbf{X} = \{x: x_{\min} \le x \le x_{\max}\}$ называется упорядоченная совокупность пар [8]

$$X = \{x, X(x)\}, \quad x \in \mathbf{X},$$

где $X(x)$ — функция принадлежности x к X, отображающая $\mathbf{X}$ в интервал [0, 1].

На нечетких множествах X_1 и X_2 определены операции объединения (связки "или", "иначе")

$$(X_1 \cup X_2)(x) = X_1(x) \vee X_2(x) = \max(X_1(x), X_2(x))$$

и пересечения (связка "и")

$$(X_1 \cap X_2)(x) = X_1(x) \wedge X_2(x) = \min(X_1(x), X_2(x)).$$

Лингвистическая переменная определяется тройкой $(x, T_x, \mathbf{X})$, в которой x — название переменной, $T_x = \{T_x^1, T_x^2, ..., T_x^k\}$ — терм-множество лингвистических значений или термов T_x^l, $l = \overline{1, k}$, с соответствующими функциями принадлежности $T_x^l(x)$, заданными на универсальном множестве $\mathbf{X}$.

Нечеткое отношение R на декартовом произведении множеств $\mathbf{X} \times \mathbf{Y} = \{(x, y), x \in \mathbf{X}, y \in \mathbf{Y}\}$ есть нечеткое множество в $\mathbf{X} \times \mathbf{Y}$ с функцией принадлежности $R(x, y)$, которая характеризует степень совместимости пары x, y с R. Если x, y — точки, т. е. $x \in \{x^1, ..., x^k\}$, $y \in \mathbf{Y} = \{y^1, ..., y^s\}$, то отношение является матрицей с элементами $R(x^l, y^r)$, $l = \overline{1, k}$, $r = \overline{1, s}$ [8].

ЛОГИКО-ЛИНГВИСТИЧЕСКИЕ РЕГУЛЯТОРЫ

Такого рода нечеткие регуляторы содержат нечеткие множества, логические операции объединения, пересечения и композиции с лингвистическими значениями переменных, нечеткое отношение, образованное одной или несколькими логическими операциями, и правило вывода нечеткого выхода при известном входе. Первые логико-лингвистические регуляторы (ЛЛР) [9—11] оказали очень сильное влияние на последующие исследования в области нечетких систем управления и заслуживают того, чтобы вначале изложить основные принципы их построения, а затем показать, как эти принципы реализованы в одном из регуляторов.

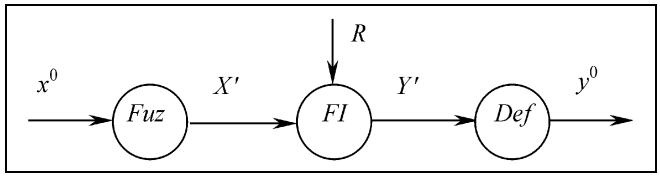


Рис. 1. Преобразование входа в нечетком регуляторе

Принципы построения ЛЛР рассмотрим на примере простейшего обобщенного регулятора с одним входом x (обычно ошибка регулирования) и одним выходом y (регулирующее или управляющее воздействие), связанных нечеткими правилами

$$\begin{aligned} &R^1\text{: если } x \text{ есть } X^1\text{, то } y \text{ есть } Y^1\text{, иначе}\\ &R^2\text{: если } x \text{ есть } X^2\text{, то } y \text{ есть } Y^2\text{, иначе}\\ &\ldots\\ &R^n\text{: если } x \text{ есть } X^n\text{, то } y \text{ есть } Y^n, \end{aligned} \qquad (1)$$

содержащими нечеткие множества $X^0 \in T_x$ и $Y^0 \in T_y$.

В алгоритме функционирования ЛЛР в той или иной форме присутствуют процедуры преобразований (фазификации *Fuz*) измеренного значения x^0 переменной x в лингвистическое значение X', нечеткого вывода (fuzzy inference *FI*) лингвистического выхода Y' по известному входу X' и совокупности правил $R = \{R^1, ..., R^n\}$ и преобразования (дефазификации *Def*) лингвистического значения выхода Y' в действительное y^0 (рис. 1).

"Входной измеримой переменной x со значением x^0 соответствует так называемое "вырожденное" нечеткое множество X' с функцией принадлежности

$$X'(x) = \begin{cases} 1, \text{ если } x = x^0 \\ 0, \text{ если } x \ne x^0, \end{cases}$$

где x^0 — точка, именуемая синглетоном множества X'. Запишем выражение нечеткого вывода для ЛЛР, заданного множеством правил (1)

$$\begin{aligned} &R^1\text{: если } x \text{ есть } X^1\text{, то } y \text{ есть } Y^1\text{, иначе}\\ &R^2\text{: если } x \text{ есть } X^2\text{, то } y \text{ есть } Y^2\text{, иначе}\\ &\ldots\\ &R^n\text{: если } x \text{ есть } X^n\text{, то } y \text{ есть } Y^n,\\ &\qquad x \text{ есть } X' \end{aligned} \qquad (2)$$

$$\overline{\qquad\qquad\qquad\qquad}$$

$$y \text{ есть } Y'$$

Значения истинности высказываний "x есть X^{θ}", "y есть Y^{θ}" и "x есть X'" в правилах (1) и посылке выражения вывода (2) определяются значениями соответствующих функций принадлежности $X^{\theta}(x)$, $Y^{\theta}(y)$ и $X'(x)$ для $x \in \mathbf{X}$, $y \in \mathbf{Y}$.

Каждое правило R^{θ} — это нечеткая импликация

$$R^{\theta} = \text{если } x \text{ есть } X^{\theta}, \text{ то } y \text{ есть } Y^{\theta} = X^{\theta} \to Y^{\theta}.$$

В ЛЛР в качестве процедуры вывода Y' используется максиминная композиция Заде [8]

$$Y'(y) = (X' \circ R)(y) = \bigvee_{x \in \mathbf{X}} (X'(x) \wedge R(x, y)), \quad (3)$$

где $R(x, y) = \bigvee\limits_{\theta = 1}^{n} R^{\theta}(x, y) = \bigvee\limits_{\theta = 1}^{n} X^{\theta}(x) \wedge Y^{\theta}(y)$.

В точке x^0 выражение (3) после подстановки $X'(x^0) = 1$ принимает вид

$$Y'(y) = \bigvee_{\theta = 1}^{n} R(x^0, y) = \bigvee_{\theta = 1}^{n} X(x^0) \wedge Y(y). \quad (4)$$

Значение выхода y^0 можно определить путем максимизации

$$y^0 = \max_{y \in Y} Y'(y) \quad (5)$$

или вычисления "центра тяжести" функции принадлежности $Y'(y)$

$$\int_{y_{\min}}^{y^0} Y'(y)\, dy = \int_{y^0}^{y_{\max}} Y'(y)\, dy. \quad (6)$$

Наиболее известный и часто цитируемый ЛЛР, разработанный для управления паровой машиной [9], имеет четыре входных (x_1 — ошибка давления, равная отклонению текущего значения от заданного значения; x_2 — ошибка скорости; x_3 — скорость изменения x_1; x_4 — скорость изменения x_2) и две выходных (y_1 — изменение тепла; y_2 — изменение давления пара) переменных. Диапазоны изменения входных $\mathbf{X_1}$, ..., $\mathbf{X_4}$ и выходной $\mathbf{Y_1}$ переменных x_1, ..., x_4, y_1 разбиваются на семь интервалов со следующими лингвистическими значениями: *PB* — положительное большое; *PM* — положительное среднее; *PS* — положительное малое; *ZE* — нулевое; *NS* — отрицательное малое; *NM* — отрицательное среднее; *NB* — отрицательное большое. Диапазон изменения $\mathbf{Y_2}$ выходной переменной y_2 состоит из пяти интервалов с определенными на них лингвистическими значениями *NB, NS, ZE, PS, PB*. Указанные лингвистические значения образуют два терм-множества $T_1 = \{NB, NM, NS, ZE, PS, PM, PB\}$ и $T_2 = \{NB, NS, ZE, PS, PB\}$. Нечеткий регулятор состоит из двух совокупностей правил, $j = 1, 2$, определяющих изменение тепла y_1 и давления y_2

R_j^1: если x_1 есть X_{1j}^1 и x_2 есть X_{2j}^1 и ...

... и x_4 есть X_{4j}^1, то y_j есть Y_j^1, иначе

R_j^2: если x_1 есть X_{1j}^2 и x_2 есть X_{2j}^2 и ...

... и x_4 есть X_{4j}^2, то y_j есть Y_j^2, иначе

. . .

$R_j^{n_j}$: если x_1 есть $X_{1j}^{n_j}$ и x_2 есть $X_{2j}^{n_j}$ и ...

... и x_4 есть $X_{4j}^{n_j}$, то y_j есть $Y_j^{n_j}$,

x_1 есть X_{1j}' и x_2 есть X_{2j}' и... и x_4 есть X_{4j}',

---

y_j есть Y_j'.

Высказывания "x_i есть $X_{ij}^{\theta_j}$" и "x_i есть X_{ij}'" в посылке выражения вывода (5) со значениями истинности, заданными соответствующими функциями принадлежности $X_{ij}^{\theta_j}(x)$ и $X_{ij}'(x_j)$, $i = \overline{1, 4}$, $j = 1, 2$, $\theta_j = \overline{1, n}$, объединены логической связкой "и", реализующей операцию пересечения. Тогда истинность левой части θ_j-го правила определяется как

$$X_{1j}^{\theta_j}(x_1) \wedge X_{2j}^{\theta_j}(x_2) \wedge ... \wedge X_{4j}^{\theta_j}(x_4),$$

а истинность посылки —

$$X_{1j}'(x_1) \wedge X_{2j}'(x_2) \wedge ... \wedge X_{4j}'(x_4), \quad j = 1, 2.$$

Выражение максиминной композиции (3) примет вид

$$Y_j'(y_j) = \bigvee_{\substack{x_1 \in X_1 \\ \vdots \\ x_4 \in X_4}} ([X_1'(x_1) \wedge (X_2'(x_2) \wedge ... \wedge X_4'(x_4)] \wedge R_j(x_1, ..., x_4, y_j)), \quad (7)$$

где $R_j(x_1, ..., x_4, y_j) = \bigvee\limits_{\theta_j = 1}^{n_j} (X_{1j}^{\theta_j}(x_1) \wedge X_{2j}^{\theta_j}(x_2) \wedge ...$

$... \wedge X_{4j}^{\theta_j}(x_4) \wedge Y_j^{\theta_j}(y_j))$.

Поскольку $x_1^0, ..., x_4^0$ являются синглетонами множеств X_i', то после подстановки $X_i'(x_i^0) = 1$,

$i = \overline{1,4}$, в выражение (7) получим импликацию Мамдани

$$Y_j'(y_j) = R_j(x_1^0, ..., x_4^0, y_j) = \bigvee_{\theta_j=1}^{n_j} (X_{1j}^{\theta_j}(x_1^0) \wedge X_{2j}^{\theta_j}(x_2^0)$$

$$\wedge ... \wedge X_{4j}^{\theta_j}(x_4^0) \wedge Y_j^{\theta_j}(y_i)), j = 1, 2. \quad (8)$$

Действительные выходные значения y_1^0 и y_2^0 определяются на основании найденных функций принадлежности $Y_1'(y_1)$ и $Y_2'(y_2)$ с помощью соотношений (5) и (6).

Рассмотренный ранее ЛЛР с импликацией (8) называется регулятором Мамдани. Если в выражении (4) принять $R(x, y) = 1$, то регулятор Мамдани будет иметь статическую характеристику многопозиционного реле (рис. 2, *а*), в котором нарушаются линейность и непрерывность выхода y относительно входа x.

Попытки устранить указанные недостатки были предприняты в работах [12—16] и заключались в использовании импликации Лукасевича в качестве нечеткого отношения $R(x, y)$ в выражении (3):

$$R_L(x, y) = 1 \wedge [1 - X(x) + Y(y)]. \quad (9)$$

Действительно, если принять $R_L(x, y) = 1$, то импликация (9) в выражении (4) с одним входом

$$Y'(y) = \bigvee_{\theta=1}^{n} R_L^{\theta}(x, y)$$

позволяет получить более совершенный ЛЛР, имеющий статическую характеристику линейной функции с насыщением (рис. 2, *б*).

Однако гораздо большее применение нашли регуляторы и нечеткие системы управления, использующие импликацию Заде. Им посвящено большое количество исследований, в которых регуляторы и системы управления представлены нечеткими дифференциальными

$$\dot{X}(t) = X(t) \circ R \quad (10)$$

и разностными уравнениями

$$X_{t+1} = X_t \circ R. \quad (11)$$

В первых публикациях [17—21] анализировались устойчивость и управляемость нечетких динамических систем типа (10) и (11). Для этих целей привлекались функции Ляпунова [17, 18] и методы оценки устойчивости, опирающиеся на такие специфические понятия нечетких множеств, как энергия нечеткого множества X_t и нечеткого отношения R, пиковые характеристики нечетких множеств и меры их близости [19—21].

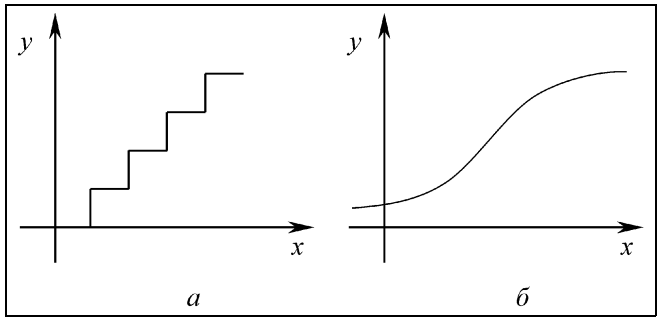


Рис. 2. Статические характеристики нечетких регуляторов

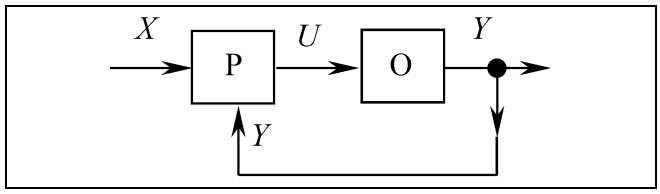


Рис. 3. Схема замкнутой системы управления

Основной недостаток предлагаемых подходов заключается в отсутствии конкретных рекомендаций по выбору или синтезу нечетких регуляторов и систем управления, обладающих определенными динамическими свойствами (управляемости, устойчивости и качества процессов регулирования).

Первая попытка синтеза оптимального в смысле минимума ошибки регулирования ЛЛР была предпринята в замкнутой системе управления (рис. 3) на основании заданных таблицами 1 и 2 нечетких операторов F_{OU} и F^* объекта [22] и оптимальной замкнутой системы, соответственно.

В целях компактности изложения представим в аналитической форме табличные операторы объекта управления O

$$\dot{Y} = F_{OU}(Y, U), \quad (12)$$

Таблица 1

Нечеткий оператор объекта

Управление U	Выход Y						
	NB	*NM*	*NS*	*ZE*	*PS*	*PM*	*PB*
NB	*ZE*	*NS*	*NS*	*NM*	*NB*	*NM*	*NS*
NM	*PB*	*ZE*	*NB*	*NB*	*NB*	*NM*	*NB*
NS	*NS*	*NS*	*ZE*	*NM*	*NM*	*NB*	*NB*
ZE	*PB*	*PB*	*PB*	*ZE*	*PS*	*PS*	*PS*
PS	*NM*	*ZE*	*PS*	*NB*	*PB*	*PM*	*PB*
PM	*NM*	*NS*	*ZE*	*PS*	*ZE*	*NS*	*NB*
PB	*NB*	*NB*	*NB*	*ZE*	*PB*	*PB*	*PB*

оптимальной замкнутой системы

$$\dot{Y} = F^*(Y, X) \qquad (13)$$

и синтезируемого регулятора P

$$U = F_{PX}(X, Y), \qquad (14)$$

в которых лингвистические переменные, характеризующие задание X, выход Y, его скорость $\dot{Y}$ и управление U принимают значения из терм-множества $T = \{NB, NM, NS, ZE, PS, PM, PB\}$. Оператор объекта (12) строится по результатам исследования его статических и динамических характеристик. Для табличного оператора объекта (12) легко получить обратный относительно управления U оператор F_{OU}^{-1}:

$$U = F_{OU}^{-1}(\dot{Y}, X), \qquad (15)$$

а оператор F^* оптимальной замкнутой системы можно получить, исходя из графика лингвистической динамики (рис. 4) и следующих эвристических соображений.

Точки 1, 2,..., 7 на графике характеризуют равенство лингвистических значений задания X и выхода Y, а также минимальную скорость выхода $\dot{Y} = ZE$, позволяющую предотвратить перерегулирование. По мере увеличения рассогласования между X и Y, т. е. ошибки регулирования, должна возрастать скорость выхода $\dot{Y}$, направленная в сторону одной из этих точек. Направление и размеры стрелок $\dot{Y}$ соответствуют принятым лингвистическим значениям. Например, в точке "∘" графика лингвистической динамики оптимальной замкнутой системы определены один набор данных табличного оператора F^*: $X = PS$, $Y = PM$, $\dot{Y} = NS$ и соответствующее правило: если $X = PS$ и $Y = PM$, то $\dot{Y} = NS$.

Таблица 2

Нечеткий оператор замкнутой системы

Вход X	Выход Y						
	NB	*NM*	*NS*	*ZE*	*PS*	*PM*	*PB*
NB	*ZE*	*NS*	*NS*	*NM*	*NB*	*NB*	*NB*
NM	*PS*	*ZE*	*NS*	*NS*	*NM*	*NB*	*NB*
NS	*PS*	*PS*	*ZE*	*NS*	*NS*	*NM*	*NB*
ZE	*PB*	*PM*	*PS*	*ZE*	*NS*	*NM*	*NB*
PS	*PB*	*PM*	*PS*	*PS*	*ZE*	*NS*	*NM*
PM	*PB*	*PB*	*PM*	*PS*	*PS*	*ZE*	*NS*
PB	*PB*	*PB*	*PB*	*PM*	*PS*	*PS*	*ZE*

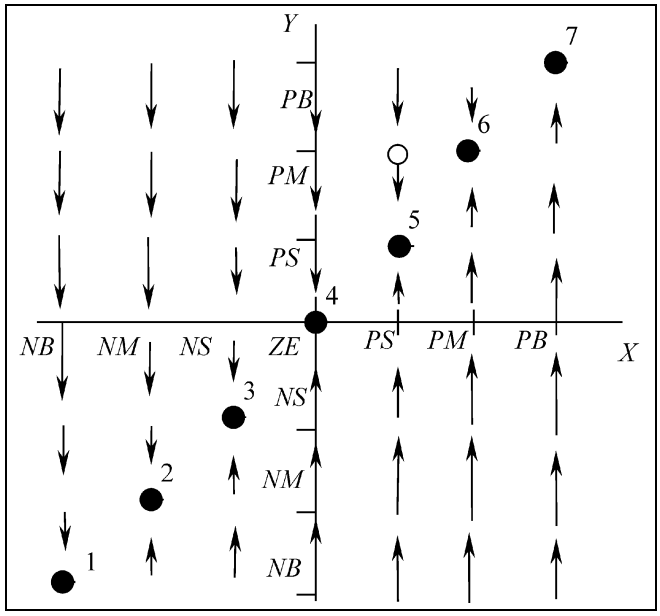


Рис. 4. График лингвистической динамики

Теперь сформулируем задачу синтеза оптимального нечеткого регулятора. Для всех лингвистических значений задания X и выхода Y, с помощью оператора оптимальной замкнутой системы (13) и обратного оператора объекта (15) определить управления U, т. е. тройки $\langle U, X, Y\rangle$, образующие оператор регулятора (14).

Рассмотрим процедуру определения управления U^* в тройке $\langle X^*, Y^*, U^*\rangle$ при $X^* = NS$, $Y^* = PM$. Подстановка $X^* = NS$ и $Y^* = PM$ в табл. 2 (оператор F^* оптимальной замкнутой системы) дает $\dot{Y}^* = NM$. Для $Y^* = PM$ и $\dot{Y}^* = NM$ из табл. 1 (оператор F_{OU} объекта управления) получаем $U^* = NM$, т. е. реализуем обратный оператор объекта и определяем искомую тройку $\langle NS, PM, NM\rangle$.

В общем случае обратный оператор F_{OU}^{-1} не является однозначным. Неоднозначным становится найденный оптимальный оператор регулятора, что значительно снижает практическую ценность такого подхода к синтезу ЛЛР.

Дальнейшее развитие методологии синтеза табличного оператора регулятора получило в работе [23]. На основании статических характеристик и переходных функций апериодического звена первого порядка, образующих операторы объекта по каналам управления F_{OU} и возмущения F_{OW} (рис. 5), а также качественного описания процесса регулирования удалось синтезировать нечеткий регулятор, действующий при изменении задания X,

$$F_P' = F_X \cup F_Y \cup F_E$$

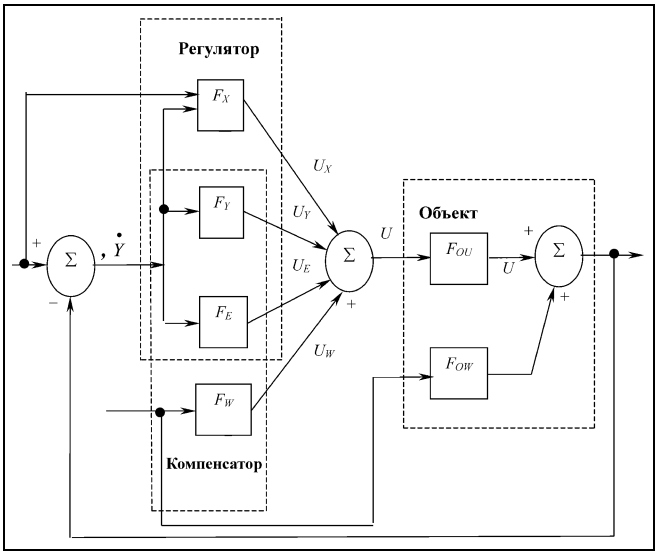


Рис. 5. Схема управления с нечетким регулятором и компенсатором

и компенсатор, устраняющий влияние возмущения W на выход,

$$F_K = F_W \cup F_Y \cup F_E.$$

Здесь $\cup$ — операция объединения компонент, реализующих три фазы управления.

На первой фазе управления $U_X = F_X(X, E)$ регулятора при существенном изменении задания X вначале значение U_X устанавливается предельным. Как только выходная величина Y достигнет некоторой окрестности задания X, по статической характеристике канала $U—Y$ выбирается управляющее воздействие, при котором установившееся значение выхода станет близким к заданию.

Управление U_W или выход составляющей $U_W = F_W(W)$ компенсатора F_K формируется, исходя из двух принципов нечеткой инвариантности. По статической характеристике канала $W—Y$ оценивается возможная реакция выхода Y_W на возмущение W и определяется управление U_W, вызывающее изменение выхода Y, равное по величине и обратное по знаку значению Y_W. Тем самым удается обеспечить частичную компенсацию возмущения или независимость (инвариантность) выхода Y от возмущения W.

Более полной компенсации можно достигнуть, если выбрать такое управление U_W, при котором скорость изменения Y_U будет равна по величине и противоположна по направлению скорости Y_W.

Компонента F_Y служит для устранения перерегулирования, а компонента F_E — статической ошибки. Исходя из предложенных принципов формирования компонентов регулятора и компенсатора, были разработаны методы синтеза табличных операторов F_X, F_W, F_Y и F_E, позволивших обеспечить требуемое качество регулирования температуры на выходе печи пиролиза ацетона [23]. Близкие подходы к синтезу табличного нечеткого регулятора были предложены для управления ректификационной установкой [24] и другими химическими объектами [25].

К основным недостаткам ЛЛР табличного типа можно отнести их ограниченную размерность (общее число переменных не должно превышать трех и субъективность выбора интервалов и соответствующих значений лингвистических переменных.

Отметим одно важное достоинство всех ЛЛР. Как было ранее сказано, ЛЛР подобен многопозиционному реле, у которого уровни срабатывания выбираются с учетом свойств объекта управления. Тем самым удается в значительной мере скомпенсировать влияние нелинейности объекта, заметно ухудшающей работу систем управления с линейными П-, ПИ- и ПИД-регуляторами.

АНАЛИТИЧЕСКИЕ РЕГУЛЯТОРЫ

Субъективность выбора интервалов и лингвистических переменных и связанное с ней снижение качества управления могут быть в значительной степени устранены в так называемых аналитических нечетких регуляторах и системах управления, работоспособность которых обеспечивается известными аналитическими и численными методами параметрической идентификации, анализа и синтеза линейных и нелинейных систем с привлечением нечетких динамических моделей.

Особое место занимает так называемая нечеткая модель Такажи и Суджено или TS-модель [26]. Сначала аналитическим путем [27], а затем в конкретных задачах моделирования и управления (в качестве регулятора) были продемонстрированы ее высокие аппроксимационные способности. Нечеткая TS-модель состоит из совокупности продукционных правил, содержащих в правой части линейные разностные уравнения [26]

$$\text{если } y(t-1) \text{ есть } Y_1^\theta, \ldots, y(t-r) \text{ есть } Y_r^\theta, \quad (16)$$

$$x(t) \text{ есть } X_0^\theta, \ldots, x(t-s) \text{ есть } X_s^\theta,$$

$$\text{то } y^\theta(t) = a_0^\theta + \sum_{k=1}^{r} a_k^\theta y(t-k) + \sum_{l=1}^{s} b_l^\theta x(t-l),$$

$$\theta = \overline{1, n},$$

где $\boldsymbol{a}^\theta = (a_0^\theta, a_1^\theta, \ldots, a_r^\theta)$, $\boldsymbol{b}^\theta = (b_0^\theta, b_1^\theta, \ldots, b_s^\theta)$ — векторы настраиваемых параметров; $\boldsymbol{y}(t-r) = (1, y(t-1), \ldots, y(t-r))$ — вектор состояния;

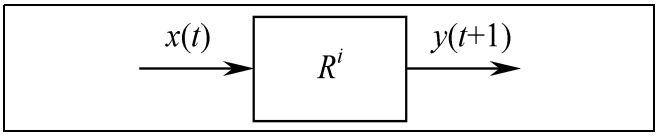


Рис. 6. Нечеткий блок

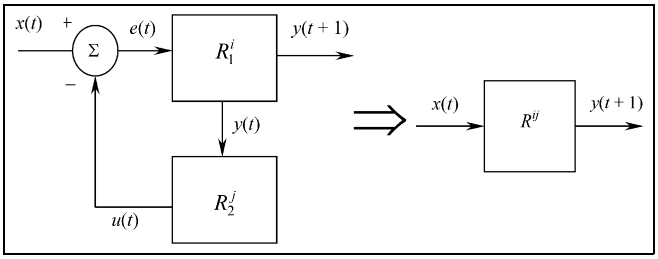


Рис. 7. Схема соединения с обратной связью и его математическая модель

$\boldsymbol{x}(t-s) = (x(t), x(t-1), ..., x(t-s))$ — входной вектор; $Y_1^\theta, ..., Y_r^\theta$; $X_0^\theta, ..., X_r^\theta$ — нечеткие множества.

Выражение (19) можно значительно упростить, если переобозначить входные переменные

$$(u_0(t), u_1(t), ..., u_m(t)) = (1, y(t-1), ..., y(t-r), x(t), ..., x(t-s)),$$

коэффициенты разностного уравнения

$$(c_0^\theta, c_1^\theta, ..., c_m^\theta) = (a_0^\theta, a_1^\theta, ..., a_r^\theta, b_1^\theta, ..., b_s^\theta)$$

и функции принадлежности

$$(U_1^\theta(u_1(t)), ..., U_m^\theta(u_m(t))) = (Y_1^\theta(y(t-1)), ..., Y_r^\theta(y(t-r)), X_0^\theta(x(t)), ..., X_s^\theta(x(t-s))),$$

где $m = r + s + 1$.

Аналитическая форма нечеткой модели (16), предназначенная для вычисления выхода $\hat{y}(t)$, имеет вид $\hat{y}(t) = \boldsymbol{c}^T\tilde{\boldsymbol{u}}(t)$, где $\boldsymbol{c} = (c_0^1, ..., c_0^n, ..., c_m^1, ..., c_m^n)^T$ — вектор уточняемых параметров; $\tilde{\boldsymbol{u}}^T(t) = (u_0(t)\beta^1(t), ..., u_0(t)\beta^\theta(t), ..., u_m(t)\beta^1(t), ..., u_m(t)\beta^n(t))$ — расширенный входной вектор; $\beta^\theta(t) = \dfrac{U_1^\theta(u_1(t)) \otimes ... \otimes U_m^\theta(u_m(t))}{\sum\limits_{\theta=1}^{N} (U_1^\theta(u_1(t)) \otimes ... \otimes U_m^\theta(u_m(t)))}$ — нечеткая функция, где $\otimes$ — операция минимизации или произведения.

При заданных в начальный момент $t = 0$ векторе $\boldsymbol{c}(0) = 0$, корректирующей матрице $Q(0)$ размером $nm \times nm$ и значениях $u(t)$ в моменты времени $t = \overline{1, N}$ вектор параметров $\boldsymbol{c}(t)$ вычисляется с помощью известного многошагового метода наименьших квадратов [27]

$$\boldsymbol{c}(t) = \boldsymbol{c}(t-1) + Q(t)\tilde{\boldsymbol{u}}(t)[y(t) - \boldsymbol{c}^T(t-1)\tilde{\boldsymbol{u}}(t)], \quad (17)$$

$$Q(t) = Q(t-1) - \frac{Q(t-1)\tilde{\boldsymbol{u}}(t)\tilde{\boldsymbol{u}}^T(t)Q(t-1)}{1 + \tilde{\boldsymbol{u}}^T(t)Q(t-1)\tilde{\boldsymbol{u}}(t)}, \quad (18)$$

$$Q(0) = \gamma I, \quad \gamma \gg 1,$$

где I — единичная диагональная матрица.

Полный алгоритм идентификации, помимо алгоритма (17), (18), содержит также алгоритмы идентификации количества правил n, порядка r, s разностного уравнения и параметров функций принадлежности [28—30].

Появление TS-модели оказало огромное влияние на последующее развитие теории нечетких систем управления:

- среди нечетких моделей для нее впервые стало правомерным применение традиционной параметрической идентификации;
- несмотря на присутствие в правой части правил линейных разностных уравнений в TS-модели, посредством уточнения параметров $\boldsymbol{c}$, порядка r, s и увеличения количества правил n удается описать с очень высокой точностью нелинейные динамические процессы;
- усредняющие свойства механизма вывода y и специфический вид функций принадлежности позволяют сделать TS-модель мало чувствительной к помехам и погрешностям измерений;
- будучи нелинейной и непрерывной функцией входных переменных и параметров, TS-модель предоставляет широкие возможности аналитического исследования устойчивости нелинейных систем с ее присутствием и последующего их обучения с целью получения требуемого качества переходных процессов.

Для замкнутой системы управления с нечетким регулятором на базе модели (16) также актуальна проблема устойчивости и ее количественной оценки. В духе классического представления линейных систем Танака и Суджено [31] предложили нечеткий блок (рис. 6) — динамический объект, описываемый нечеткой разностной моделью (16) в векторной форме

$$R^i: \text{если } \boldsymbol{y}(t) \text{ есть } \boldsymbol{Y}^i \text{ и } \boldsymbol{x}(t) \text{ есть } \boldsymbol{X}^i,$$

$$\text{то } y^i(t+1) = a_0^i + \sum_{k=1}^{r} a_k^i y(t-k+1) + \sum_{l=1}^{s} b_l x(t-l+1), \quad (19)$$

где $\boldsymbol{y}(t) = [y(t), y(t-1), ..., y(t-r+1)]^T$, $\boldsymbol{x}(t) = [x(t), x(t-1), ..., x(t-s+1)]^T$; $Y^i = [Y_1^i, ..., Y_r^i]$, $X^i = [X_1^i, ..., X_s^i]$; r, s — порядок разностного урав-

нения; $\boldsymbol{y}(t)$ есть $\boldsymbol{Y}^I \Rightarrow y(t)$ есть Y_1^i и... и $y(t-r+1)$ есть Y_r^i.

Из таких блоков формируются различные соединения (параллельные и с обратной связью) и выводятся их математические модели.

Например, соединение с обратной связью (рис. 7), содержащее блоки объекта

$$R_1^i: \text{ если } \boldsymbol{y}(t) \text{ есть } Y_1^i \text{ и } \boldsymbol{e}(t) \text{ есть } E_1^i,$$

$$\text{то } y^i(t+1) = a_{10}^i + \sum_{k=1}^{r} a_{1k}^i y(t-k+1) + \sum_{l=1}^{s} b_{1l}^i e(t-l+1)$$

и регулятора

$$R_2^i: \text{ если } \boldsymbol{y}(t) \text{ есть } Y_2^j \text{ и } \boldsymbol{e}(t) \text{ есть } E_2^j, \quad (20)$$

$$\text{то } u^j(t) = a_{20}^j + \sum_{k=1}^{r} a_{2k}^j y(t-k+1)$$

эквивалентно блоку

$$R^{ij}: \text{ если } \boldsymbol{y}(t) \text{ есть } \boldsymbol{Y}^{ij} \text{ и } \boldsymbol{e}(t) \text{ есть } E^{ij},$$

$$\text{то } y^{ij}(t+1) = a_{10}^i - b_1^i a_{20}^j + b_1 x(t) + \sum_{k=1}^{r} (a_{1k} - b_1^i a_{2k}^j) y(t-k+1),$$

где $i = 1, 2, ..., n_1$, $j = 1, 2, ..., n_2$; $\boldsymbol{e}(t) = [x(t) - u(t), x(t-1) - u(t-1), ..., x(t-m+1) - u(t-m+1)]^T$; $Y^{ij} = (Y_1^i \cap Y_2^j)$, $E^{ij} = (E_1^i \cap E_2^j)$.

Вывод аналитических оценок устойчивости нечетких систем (19) и (20) осуществляется с помощью метода Ляпунова на основании уравнения свободного движения

$$R^i: \text{ если } \boldsymbol{y}(t) \text{ есть } Y_1^i \text{ и } \boldsymbol{y}(t-r+1) \text{ есть } Y_r^i, \quad (21)$$

$$\text{то } y^i(t+1) = a_1^i y(t) + ... + a_r^i y(t-r+1),\ i = \overline{1, n},$$

правую часть которого можно записать в матричной форме $A_i \boldsymbol{y}(t)$, где $y(t) = [y(t), y(t-1), ..., y(t-r+1)]^T$,

$$A_i = \begin{bmatrix} a_1^i & a_2^i & ... & a_{r-1}^i & a_r^i \\ 1 & 0 & ... & 0 & 0 \\ 0 & 1 & ... & 0 & 0 \\ 0 & 0 & ... & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & ... & 0 & 0 \\ 0 & 0 & ... & 1 & 0 \end{bmatrix}.$$

В работах [31—33] показано, что нечеткая система (21), представленная расчетной зависимостью

$$y(t+1) = \sum_{i=1}^{n} w^i A_i \boldsymbol{y}(t) / \sum_{i=1}^{n} w^i,$$

является асимптотически устойчивой в глобальном смысле, если для всех подсистем существует положительно определенная матрица B такая, что

$$A_i^T B A_i - B < 0,\ \forall i \in \{1, 2, ..., n\}.$$

Справедливость этой оценки была подтверждена лишь для самого простого пропорционального регулятора.

Близкий подход к анализу устойчивости, опирающийся на методы Ляпунова, был развит в работе [34] для нечеткой системы в пространстве состояний

$$R^i: \text{ если } \boldsymbol{y}_1(t) \text{ есть } \boldsymbol{Y}_1^i, ..., \boldsymbol{y}_r(t) \text{ есть } \boldsymbol{Y}_r^i,$$

$$\text{то } x(t) \text{ есть } X^i, \text{ то}$$

$$y_1(t+1) = a_{11}^i y_1(t) + a_{12}^i y_2(t) + ... + a_{1r}^i y_r(t) + b_1^i x(t),$$

$$\cdots$$

$$y_1(t+1) = a_{r1}^i y_1(t) + a_{r2}^i y_2(t) + ... + a_{rr}^i y_r(t) + b_r^i x(t)$$

и получена аналитическая оценка устойчивости замкнутой системы с пропорциональным регулятором. Для достижения устойчивости предлагается методом градиентов уточнять параметры a_{jl}^i и коэффициент усиления регулятора.

Аналогичные подходы к анализу устойчивости нечетких систем, использующие методы Ляпунова с последующим синтезом регуляторов, изложены в работах [35—48]. Ограниченность метода Ляпунова очевидна: он позволяет реализовать устойчивость системы управления лишь самыми простыми пропорциональными регуляторами и не дает рекомендаций по достижению требуемого качества переходных процессов. Функции поддержания качества переходных процессов в нечетких системах управления могут обеспечить нечеткие обучаемые регуляторы и системы управления.

ОБУЧАЕМЫЕ НЕЧЕТКИЕ РЕГУЛЯТОРЫ

Такого рода регулятор должен обладать способностью приобретать знания о поведении объекта и системы и на их основе вырабатывать управление, при котором ошибка регулирования не превышает допустимой величины. В процессе обучения, помимо регулятора, участвует модель объекта, которая также должна приобретать знания и настраиваться

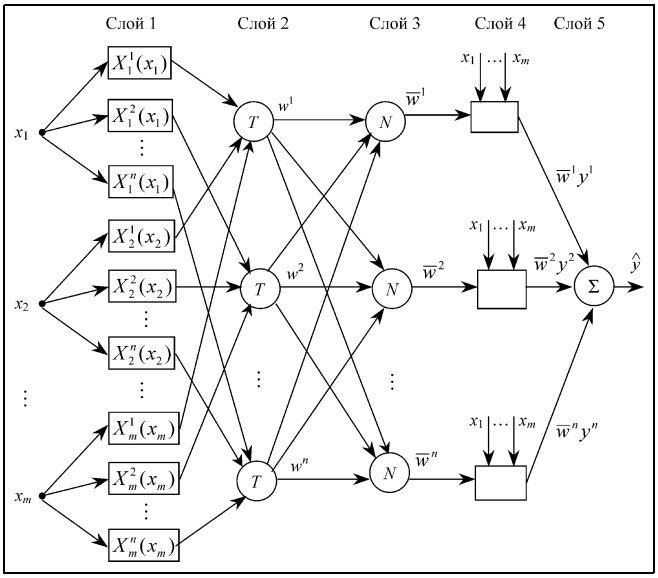


Рис. 8. Нейронечеткая структура TS-модели

на меняющиеся условия функционирования объекта. Указанным требованиям и описанию регулятора и объекта управления в достаточно полной мере удовлетворяет нечеткая TS-модель со структурой пятислойной нейронной сети прямого действия, известной под именем ANFIS (Adaptive-Network-based Fuzzy Inference Systems) [49—51].

Нейронечеткая структура обобщенной TS-модели с n правилами и m входами, реализующая механизм вывода y, изображена на рис. 8.

В первом слое при $x_i = x_i^0$, $i = \overline{1, m}$, вычисляются степени функций принадлежности, а во втором слое они обрабатываются T-оператором минимизации или произведения. В третьем слое (N) определяются нормализованные веса $\overline{w}^{\theta} = w^{\theta}/(w^{\theta} + w^{\theta})$, $\theta = \overline{1, n}$, на которые в четвертом слое умножаются соответствующие значения y^{θ}, найденные по линейным уравнениям

$$y^1 = b_0^1 + b_1^1 x_1 + b_2^1 x_2 + ... + b_m^1 x_m,$$
$$y^2 = b_0^2 + b_1^2 x_1 + b_2^2 x_2 + ... + b_m^2 x_m,$$
$$\cdots$$
$$y^n = b_0^n + b_1^n x_1 + b_2^n x_2 + ... + b_m^n x_m.$$

Пятый слой — суммирование и получение итогового значения $\hat{y}$.

Приобретение TS-моделью знаний об объекте заключается в определении коэффициентов уравнений b_i^{θ}, $i = \overline{0, m}$, $\theta = \overline{1, n}$, параметров функций принадлежности $\boldsymbol{d}_i^{\theta}$, $i = \overline{1, m}$, и числа правил n, при которых выходы модели $\hat{y}$ и объекта y совпадают или становятся близкими.

Если число правил n фиксировано, функции принадлежности $\boldsymbol{X}_i^{\theta}(x_i, \boldsymbol{d})$ непрерывны относительно параметров $\boldsymbol{d}$ и обрабатываются T-оператором произведения ($w^{\theta} = \prod_{i=1}^{m} \boldsymbol{X}_i^{\theta}(x_i)$, $\theta = \overline{1, n}$), то TS-модели можно обучать методом обратного распространения ошибки BP (Back Propagation), предложенным в работах [52, 53]. Он заключается в минимизации квадратической ошибки $I = 0{,}5(y - \hat{y}(\boldsymbol{c}))^2$ градиентным методом

$$\boldsymbol{c}^{\lambda+1} = \boldsymbol{c}^{\lambda} - h\frac{\partial I}{\partial \boldsymbol{c}}, \qquad (22)$$

где $\boldsymbol{c} = (\boldsymbol{b}, \boldsymbol{d})$ — вектор параметров; h — рабочий шаг.

На основании цепного правила определения частных производных по $\boldsymbol{d}_l^{\theta}$

$$\frac{\partial I}{\partial \boldsymbol{d}_l^{\theta}} = \frac{\partial I}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial w^{\theta}} \cdot \frac{\partial w^{\theta}}{\partial X_l^{\theta}} \cdot \frac{\partial X_l^{\theta}}{\partial \boldsymbol{d}_l^{\theta}}$$

и по b_l^{θ} $\quad \dfrac{\partial I}{\partial b_l^{\theta}} = \dfrac{\partial I}{\partial \hat{y}} \cdot \dfrac{\partial \hat{y}}{\partial b_l^{\theta}}$

без промежуточных выкладок запишем их аналитические выражения

$$\frac{\partial I}{\partial \boldsymbol{d}_l^{\theta}} = (y - \hat{y}) - \frac{y^{\theta} - w^{\theta}\hat{y}}{\sum_{j=1}^{2} w^j} \left(\prod_{\substack{i=1 \\ i \neq l}}^{m} X_i^{\theta}(x_i) \right) \frac{\partial X_l^{\theta}}{\partial \boldsymbol{d}_l^{\theta}},$$
$$l = \overline{1, m},$$

$$\frac{\partial I}{\partial b_l^{\theta}} = (y - \hat{y}) x_l,\ l = \overline{1, m},\ \theta = \overline{1, n},\ x_0 = 1.$$

В качестве функции принадлежности обычно выбираются сигмоидные $X(x) = (1 + \exp(dx))^{-1}$, $d > 0$, или радиально-базисные $X(x) = \exp(d_1(x - d_2))$ функции, дифференцируемые по d, d_1 и d_2.

Приведем постановку задачи обучения нейронечеткого регулятора P прямого действия, т. е. соединенного последовательно с объектом O (рис. 9).

Пусть объект управления с одним выходом описывается дискретным уравнением

$$y(t + 1) = f_0(y(t), ..., y(t - r), u(t), ..., u(t - s)) \quad (23)$$

порядка r, s.

Предположим, что объект управления обратимый, т. е. существует функция f_0^{-1}, инверсная уравнению (23):

$$u(t) = f_0^{-1}(y(t+1), y(t), ..., y(t-r), u(t-1), ..., u(t-r)).$$

Рассмотрим нейронечеткую TS-модель (см. рис. 8) с m-мерным входным вектором $\boldsymbol{x}_\text{р}(t) = (y(t+1), y(t), ..., y(t-r), u(t-1), ..., u(t-r))$:

$$\hat{u}(t) = f_\text{р}(x_\text{р}(t), c_\text{р}),$$

в которой обеспечивается требуемая близость $\hat{u}(t)$ к $u(t)$ при соответствующих входах и которая предлагается в качестве регулятора. Обучение регулятора алгоритмом *АОР* управлению с минимальной квадратической ошибкой

$$I_\text{р} = 0{,}5(x(t+1) - y(t+1))^2 = 0{,}5e_\text{р}^2(t+1)$$

в последовательной схеме (см. рис. 9) связано с вычислением выражения

$$\frac{\partial I_\text{р}}{\partial \boldsymbol{c}_\text{р}} = \frac{\partial I_\text{р}}{\partial y} \cdot \frac{\partial y}{\partial u} \cdot \frac{\partial u}{\partial c_\text{р}},$$

в котором не определено значение производной $\partial y/\partial u$. Его можно легко найти с помощью нейронечеткой модели объекта, именуемой эмулятором

$$\hat{y}(t+1) = f_\text{э}(\hat{y}(t), ..., \hat{y}(t-r), u(t), ..., u(t-r), c_\text{э}),$$

вычисляя $\partial\hat{y}/\partial u$ вместо $\partial y/\partial u$. Обучение эмулятора *Э* алгоритмом *АОЭ*, обеспечивающим минимальную квадратическую ошибку

$$I_\text{э} = 0{,}5(y(t+1) - \hat{y}(t+1))^2 = 0{,}5e_\text{э}^2(t+1),$$

также как и регулятора, осуществляется ВР-методом [54—57].

Все основные недостатки первых обучаемых систем управления связаны с применением ВР-метода, а именно: локальный характер поиска и частое «зацикливание», присущие градиентным методам; требование непрерывности и дифференцируемости функций принадлежности; не определяются порядок (r, s) и количество правил n.

Для устранения "зацикливания" в работах [58—63] рекомендуется генетическим алгоритмом менять (встряхивать) размер рабочего шага h в формуле градиентов (22) и компоненты вектора $\boldsymbol{c}$. Другие исследователи [64—66] считают, что для целей обучения достаточно применять только генетические алгоритмы, позволяющие преодолеть первые два недостатка ВР-метода.

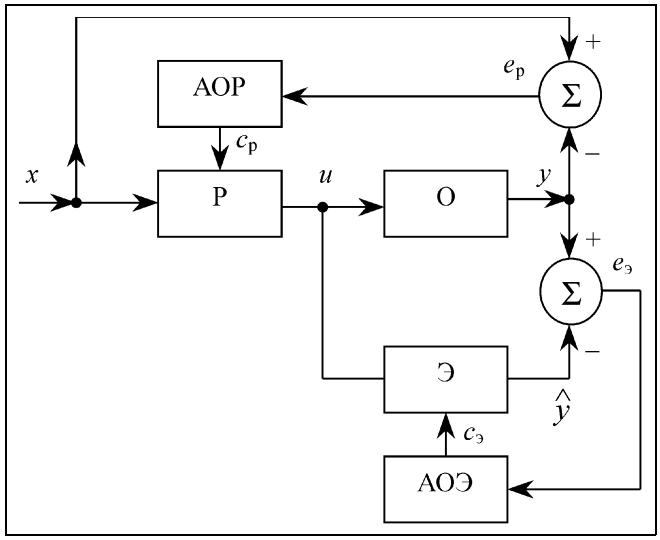


Рис. 9. Последовательная схема управления

Наибольший эффект был достигнут при гибридном обучении, осуществляемом генетическим алгоритмом, уточняющим параметры функций принадлежности $\boldsymbol{d}$, совместно с многошаговым методом наименьших квадратов (17), (18) для нахождения вектора $\boldsymbol{b}$ и другими алгоритмами, определяющими порядок r, s и число правил n в TS-модели [67]. При таком подходе удалось в значительной мере устранить все недостатки ВР-метода.

Обучаемые нечеткие регуляторы и системы управления относятся к классу наиболее перспективных. Они сохраняют высокую работоспособность в условиях помех и погрешностей измерения и достаточно быстро настраиваются на меняющиеся условия производства, снижая тем самым потери от неэффективного управления.

Вместе в тем, в ряде случаев и ЛЛР показывают высокую работоспособность. Так, в системах управления достаточно простыми объектами ЛЛР могут успешно конкурировать с обучаемыми регуляторами.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ РАБОТЫ НЕЧЕТКИХ РЕГУЛЯТОРОВ, СИНТЕЗИРОВАННЫХ МЕТОДАМИ НЕЧЕТКОЙ ЛОГИКИ И ТЕОРИИ ВЕРОЯТНОСТЕЙ

Несмотря на достигнутые результаты в теории и применении нечетких регуляторов, всё ещё есть много вопросов, требующих своего разрешения. Это относится к обоснованию и выбору метода синтеза регулятора и оценке преимуществ синтеза нечетких регуляторов по сравнению с классическими методами синтеза, например, с вероятностными методами. В основе синтеза нечетких регуляторов лежат суждения и опыт эксперта. Между тем, и в теории вероятностей для оценки количе-

ственной меры уверенности эксперта используются субъективные вероятности [68, 69]. Естественно, что вероятностные методы синтеза имеют сегодня более солидную математическую основу, чем методы нечеткой логики, основанные на выборе функции принадлежности и определении базовой лингвистической переменной [70].

В работах [70—73] исследуются вопросы построения нечетких ПИ- и ПИД-регуляторов и сравнения с классическими вероятностными методами синтеза регуляторов. Рассмотрим построение нечеткого линейного ПИ-регулятора с двумя входами. Таблица, с помощью которой реализуется ПИ-алгоритм, имеет вид табл. 3 [71].

Интегрирующий исполнительный механизм формирует управляющее воздействие по формуле $u_k = u_{k-1} + \Delta u_k$. Нормированные значения отклонения регулируемой величины и её приращения $y = 0{,}2$; $\Delta y = 0{,}9$. Этим значениям входных сигналов соответствуют четыре лингвистические переменные

$$Y = ZE;\ Y_2 = PS;\ \Delta Y_1 = PM;\ \Delta Y_2 = PB \qquad (24)$$

со следующими значениями функций принадлежности:

$$\mu_1(\varepsilon) = 0{,}4;\ \mu_2(\varepsilon) = 0{,}6;\ \mu_1(\Delta\varepsilon) = 0{,}3;\ \mu_2(\Delta\varepsilon) = 0{,}7.$$

Результаты работы [72] позволили получить рейтинги регулирующих воздействий [70]

$$\Delta U_1 = PS;\ C_1(\Delta u_1) = \min[\mu_1(\varepsilon),\ \mu_1(\Delta\varepsilon)] = 0{,}3;$$
$$\Delta U_2 = PM;\ C_2(\Delta u_2) = \min[\mu_1(\varepsilon),\ \mu_2(\Delta\varepsilon)] = 0{,}4;$$
$$\Delta U_3 = PM;\ C_3(\Delta u_3) = \min[\mu_2(\varepsilon),\ \mu_1(\Delta\varepsilon)] = 0{,}3;$$
$$\Delta U_4 = PM;\ C_4(\Delta u_4) = \min[\mu_2(\varepsilon),\ \mu_2(\Delta\varepsilon)] = 0{,}6.$$

Таким образом, регулирующее воздействие определяется двумя лингвистическими переменными *PS* и *PM* с рейтингом 0,3 и 1,3, соответственно. Значения приращения регулирующего воздействия представляет собой абсциссу x_0 центра тяжести фигуры, состоящей из двух треугольников, опирающихся на диапазоны лингвистических переменных *PS* и *PM* с высотами, равными 0,3 и 1,3, соответственно: $x_0 = 0{,}603$.

Таблица 3

Оператор нечеткого линейного ПИ-регулятора

y/Δ*y*	*NB*	*NM*	*NS*	*ZE*	*PS*	*PM*	*PB*
NB	*NB*	*NB*	*NM*	*NM*	*NS*	*NS*	*ZE*
NM	*NB*	*NM*	*NM*	*NS*	*NS*	*ZE*	*PS*
NS	*NM*	*NM*	*NS*	*NS*	*ZE*	*PS*	*PS*
ZE	*NM*	*NS*	*NS*	*ZE*	*PS*	*PS*	*PM*
PS	*NS*	*NS*	*ZE*	*PS*	*PS*	*PM*	*PM*
PM	*NS*	*ZE*	*PS*	*PS*	*PM*	*PM*	*PB*
PB	*ZE*	*PS*	*PS*	*PM*	*PM*	*PB*	*PB*

В общем случае полученное число следует умножить на диапазон изменения приращения регулирующего воздействия.

В работах [70, 73, 74] рассмотрен синтез нечеткого регулятора на основе вероятностных методов. Для принятых выше значений $y = 0{,}2$; $\Delta y = 0{,}9$ лингвистические переменные определяются также формулами (24). Но вместо функций принадлежности им соответствуют следующие значения плотностей распределения:

$$p_1(\varepsilon) = 1{,}2;\ p_2(\varepsilon) = 1{,}8;\ p_1(\Delta\varepsilon) = 0{,}9;\ p_2(\Delta\varepsilon) = 2{,}1.$$

Лингвистические переменные приращения регулирующего воздействия формируется согласно табл. 3. Значения плотности распределения лингвистических переменных приращения регулирующего воздействия в этом случае:

$$\Delta U_1 = PS;\ p_1(\Delta u) = 1,\ 08;$$
$$\Delta U_2 = PM;\ p_2(\Delta u) = 2{,}52;$$
$$\Delta U_3 = PM;\ p_3(\Delta u) = 1{,}62;$$
$$\Delta U_4 = PM;\ p_4(\Delta u) = 3{,}78.$$

Приращение регулирующего воздействия в этом случае равно математическому ожиданию (центру тяжести) найденных лингвистических переменных: $\Delta u = 0{,}627$.

Таким образом, для данного примера результаты синтеза нечеткого регулятора методами нечетких множеств и вероятностными методами практически совпадают.

Правда, надо признать, что в исследуемом примере методы теории вероятностей логически более обоснованны и не требуют специальных эвристических процедур.

ЗАКЛЮЧЕНИЕ

Современные технологические и социально-экономические системы представляют собой сложные слабоформализуемые системы, функционирующие в условиях большой неопределенности, неполноты знаний и нечеткости описаний как самой системы, так и действующих на неё сигналов [75]. Со временем стало ясно, что для управления такими системами уже недостаточно применения классических методов теории управления и необходима разработка новых методов и подходов. Один из таких подходов базируется на нечетких множествах и нечеткой логике Л. Заде. Вначале этот подход применялся и показал свою эффективность при создании экспертных систем. Несколько позднее он стал применяться для создания систем экспертного управления, а в последнее время — для синтеза регуляторов и систем управления технологическими системами. В этой статье рассмотрены, в основном, три класса нечетких регуляторов: логико-лингвистические, аналитические и обучаемые.

Отметим, что ряд теоретических и прикладных исследований не укладывается в предложенную классификацию. Так, например, системы управления, основанные на знаниях, сочетают в себе алгоритмы, использующие, помимо экспертных знаний, аналитические и конструкторские знания об объекте управления и действующих сигналах, о трендах изменений параметров объекта и т. п. [75]. С другой стороны, развиваются методы построения нечетких регуляторов на основе современных тенденций в теории управления. Например, на основе идей систем с переменной структурой [76], многомерных, многосвязных систем [77], обучающихся градиентных и нейронных алгоритмов [78] и др.

Выяснилась связь между классическими методами синтеза регуляторов и нечеткими регуляторами [70, 73]. В ряде случаев классические подходы дают возможность улучшить теоретическое обоснование построения нечетких алгоритмов. Нечеткие регуляторы доказали свою эффективность при управлении статическими объектами и постепенно завоевывают все более широкие области применения в управлении динамическими системами. Они сохраняют работоспособность в условиях помех и погрешностей измерения и достаточно быстро учитывают и настраиваются на меняющиеся условия производства, снижая тем самым потери от неэффективного управления.

Авторы надеются, что данная работа, направленная на конструктивный анализ имеющихся публикаций по нечетким регуляторам, поможет читателям ориентироваться в существующих подходах к разработке нечетких систем управления и их применениях в различных отраслях.

ЛИТЕРАТУРА

1. *Алиев Р. А., Абдикеев Н. М., Шахназаров М. М.* Производственные системы с искусственным интеллектом. — М.: Радио и связь, 1990. — 264 с.
2. *Кудинов Ю. И.* Нечеткие системы управления // Изв. АН СССР. Техническая кибернетика. — 1990. — № 5. — С. 196—206.
3. *Кузьмин В. Б., Травкин С. И.* Теория нечетких множеств в задачах управления и принципах устройства нечетких процессоров: Обзор зарубежной литературы // Автоматика и телемеханика. − 1992. − № 11. − С. 3—36.
4. *Захаров В. Н., Ульянов С. В.* Нечеткие модели интеллектуальных промышленных регуляторов и систем управления. I. Научно-организационные, технико-экономические и прикладные аспекты // Изв. РАН. Техническая кибернетика. — 1992. — № 5. — С. 171—196.
5. *Захаров В. Н., Ульянов С. В.* Нечеткие модели интеллектуальных промышленных регуляторов и систем управления. II. Эволюция и принципы построения // Изв. РАН. Техническая кибернетика. — 1993. — № 4. — С. 189—205.
6. *Захаров В. Н., Ульянов С. В.* Нечеткие модели интеллектуальных промышленных регуляторов и систем управления. III. Методология проектирования // Изв. РАН. Техническая кибернетика. — 1993. — № 5. — С. 197—220.
7. *Захаров В. Н., Ульянов С. В.* Нечеткие модели интеллектуальных промышленных регуляторов и систем управления. IV. Имитационное моделирование // Изв. РАН. Техническая кибернетика. — 1994. — № 5. — С. 169—211.
8. *Заде Л. А.* Понятие лингвистической переменной и его применение к принятию приближенных решений. — М.: Мир, 1976. — С. 165.
9. *Mamdani E. H., Assilian S.* An experiment in linguistic synthesis with a fuzzy logic controller // Int. J. Man-Machine Studies. — 1975. — No. 7. — P. 1—13.
10. *Kickert W. J. M., Van Nauta Lemke H. R.* Application of fuzzy controller in a warm water plant // Automatica. — 1976. — No. 12. — P. 301—308.
11. *King P. J., Mamdani E. H.* Application of fuzzy control system to industrial processes // Automatica. — 1977. — No. 13. — P. 235—242.
12. *Giles R.* Lukasiewicz logic and fuzzy set theory // Int. J. Man-Machine Studies. — 1976. — No. 8. — P. 313—327.
13. *Braae M., Rutherford D. A.* Fuzzy relations in a control setting // Kybernetes. — 1978. — No. 7. — P. 185—188.
14. *Kickert W. J. M., Mamdani E.H.* Analysis of a fuzzy logic controller // Fuzzy Sets and Systems. — 1978. — No. 1. — P. 29—44.
15. *Pedrycz W.* On the use of fuzzy Lukasiewicz logic for fuzzy control // Archiwum automatyki i telemechaniki. — 1980. — Vol. 25, No. 3. — P. 301—314.
16. *Baldwin J. F., Guild N. C. F.* Modelling controllers using fuzzy relations // Kybernetes. — 1980. — No. 3. — Vol. 9. — P. 223—229.
17. *Glass M.* Theory of fuzzy systems // Fuzzy Sets and Systems. — 1983. — Vol. 10. — P. 65—77.
18. *Glass M.* Invariance and stability of fuzzy systems // J. Math. Analysis and Appl. — 1984. — Vol. 99, No.1. — P. 299—319.
19. *Kiszka I. B., Gupta M. M., Nikiforuk P. N.* Energetistic stability of fuzzy dynamic systems // IEEE Trans. Syst. Man and Cybern. — 1985. — Vol. SMC-15, No. 5. — P. 783—792.
20. *Jain R.* Outline of an approach for the analysis of fuzzy systems // Int. J. Control. — 1976. — Vol. 27, No. 3. — P. 627—640.
21. *Tong R. M.* Analysis and control of fuzzy systems using finite discrete relations // Int. J. Control. — 1978. — Vol. 27, No. 3. — P. 431—440.
22. *Braae M., Rutherford D. A.* Theoretical and linguistic aspects of the fuzzy controller // Automatica. — 1979. — Vol. 12. — P. 553—577.
23. *Кудинов Ю. И.* Синтез нечеткой системы управления // Известия РАН. Теория и системы управления. — 1999. — № 1. — С. 166—172.
24. *Кудинов Ю. И., Дорохов И. Н.* Управление ректификационной установкой на основе нечетких множеств // Теоретические основы химической технологии. — 1991. — Т. 25, № 4. — 563—569 с.
25. *Кудинов Ю. И., Дорохов И. Н.* Новый принцип построения регуляторов для сложных химико-технологических объектов на основании качественной информации // Докл. РАН. — 1994. Т. 336, № 1. — С. 75—79.
26. *Takagi T., Sugeno M.* Fuzzy identification of systems and its applications to modeling and control. // IEEE Trans. Systems Man Cybernet. — 1985. — Vol. 15, No. 116. — P. 116—132.
27. *Buckley J. J.* Sugeno-type controller are universal controllers // Fuzzy Sets and Systems. — 1993. — No. 53. — P. 299—303.
28. *Кудинов Ю. И., Венков А. Г., Келина А. Ю.* Моделирование технологических и экологических процессов. − Липецк: ЛЭГИ, 2001. — С. 131.
29. *Венков А. Г.* Построение и идентификация нечетких математических моделей технологических процессов в условиях неопределенности / Дисс. канд. техн. наук, Липецк: ЛГТУ, 2002. — 147 с.
30. *Ali Y. M., Zhang L.* A methodology for fuzzy modeling of engineering systems // Fuzzy Sets and Systems. — 2001. — No. 118. — P. 181—197.
31. *Tanaka K., Sugeno M.* Stability analysis and design of fuzzy control systems // Fuzzy Sets and Systems. — 1992. — No. 45. — P. 135—156.

32. *Tanaka K., Sano M.* Fuzzy stability criterion of a cross of nonlinear systems // Inform. Sci. — 1993. — No. 71. — P. 3—26.
33. *Tanaka K., Sano M.* A robust stabilization problem of fuzzy controller systems and its applications to backing up control of a truck-trailer // IEEE Trans. Fuzzy Systems. — 1994. — No. 2. — P. 1—13.
34. *Kim W. C., Ahn S. C., Kwon W. H.* Stability analysis and stabilization of fuzzy state space models // Fuzzy Sets and Systems. — 1995. — No. 71. — P. 131—142.
35. *Wang Y. A., Wang H. C.* On stability of a fuzzy systems model // Control Theory Appl. — 1995. — No. 12. — P. 335—341.
36. *Kim C. W., Cho Y. W., Park M.* A multirule-based controller using the robust property of a fuzzy controller and its design method // IEEE Trans. Fuzzy Systems. — 1996. — No. 4. — P. 315—327.
37. *Wang H. O., Tanaka K., Griffin M. F.* An approach to fuzzy control of nonlinear systems: stability and design issue // IEEE Trans. Fuzzy Systems. — 1996. — No. 4. — P. 14—23.
38. *Tanaka K., Ikeda T., Wang H. O.* Robust stabilization of a cross of uncertain nonlinear systems via fuzzy control: quadratic stabilizability, H· control theory and linear matrix inequalities // IEEE Trans. Fuzzy Systems. — 1996. — No. 4. — P. 1—13.
39. *Ying H.* Analytical study on structure, stability and design of general nonlinear Takagi-Sugeno fuzzy control systems // Automatica. — 1998. — Vol. 34, No. 12. — P. 1617—1623.
40. *Tanaka K., Ikeda T., Wang H. O.* Fuzzy regulators and fuzzy observers: relaxed stability conditions and LMI-based designs // IEEE Trans. Fuzzy Systems. — 1998. — Vol. 6, No. 12. — P. 1—16.
41. *Yoneyama J., Nishikawa M., Katayama H.* Output stabilization of Takagi-Sugeno fuzzy systems // Fuzzy Sets and Systems. — 2000. — No. 111. — P. 253—266.
42. *Ma X.-J., Sun Z.-O.* Output tracking and regulation of nonlinear system based on Takagi-Su-geno fuzzy model // IEEE Trans. Syst. Man Cybern. — Part B: Cybern. — 2000. — Vol. 30, No. 1. — P. 47-59.
43. *Chou J.-H., Chen S.-H.* Stability analysis of the discrete Takagi-Sugeno fuzzy model with time-varying consequen usertainties // Fuzzy Sets and Systems. — 2001. — No. 118. — P. 271—279.
44. *Zang J.-M., Li R.-H., Zhang P.-A.* Stability analysis and systematic design of fuzzy control systems // Fuzzy Sets and Systems. — 2001. — No. 120. — P. 65—72.
45. *Guerra T. M., Vermeiren L.* Control laws for Takagi-Sugeno fuzzy models // Fuzzy Sets and Systems. — 2001. — No. 120. — P. 95—108.
46. *Yoneyama J., Nishikawa M., Katayama H., Ichikawa A.* Design of output feedback controllers for Takagi-Sugeno fuzzy systems // Fuzzy Sets and Systems. — 2001. — No. 121. — P. 127—148.
47. *Sun Q., Li R., Zhang P.* Stable and optimal adaptive fuzzy control of complex systems using fuzzy dynamic model // Fuzzy Sets and Systems. — 2003. — No. 133. — P. 1—17.
48. *Chang W.-J., Sun C.-C.* Constrained fuzzy controller design of discrete Takagi-Sugeno fuzzy model // Fuzzy Sets and Systems. — 2003. — No. 133. — P. 37—55.
49. *Takagi H., Hayashi I.* NN-driven fuzzy reasoning // Int. J. Appr. Reasoning. — 1991. — No. 5. — P. 191—212.
50. *Yager R. R.* Implementing fuzzy logic controllers using a neural network framework // Fuzzy Sets and Systems. — 1992. — No. 48. — P. 53—64.
51. *Bukcley J. J., Hayashi I.* Neural nets for fuzzy systems // Fuzzy Sets and Systems. — 1995. — No. 71. — P. 265—276.
52. *Lin C. T., Lee C. S. G.* Neural-network-based fuzzy logic control and decision system // IEEE Trans. Comput. — 1991. — Vol. 40, No. 12. — P. 1320—1336.
53. *Jang J.-S. R.* ANFIS Adaptive-network-based fuzzy system // IEEE Trans. Systems Man Cybernet. — 1993. — Vol. 23, No. 6. — P. 665—685.
54. *Bukcley J. J., Hayashi I., Czogala E.* Fuzzy neural controller // Proc. IEEE Int. Conf. on Fuzzy Systems. San Diego. — 1992. — P. 197—202.
55. *Tang J. R., Sun C.* Neuro-fuzzy modeling and control // Proc. IEEE. — 1995. — Vol. 83, No. 3. — P. 378—406.
56. *Shi Y., Mizumoto M., Yubazaki N., Otani M.* A method of fuzzy rules generation based on neuro-fuzzy learning algorithm // J. Japan Soc. Fuzzy Theory Systems. — 1996. — Vol. 8, No. 4. — P. 695—705.
57. *Juang C. F., Lin C.-T.* An on-line self-constructing neural fuzzy inference network and its applications // IEEE Trans. on Fuzzy Systems. — 1998. — Vol. 5, No. 1. — P. 12—32.
58. *Ishigami H., Fukuda T., Shibata T., Arai F.* Structure optimization of fuzzy neural network by genetic algorithm // Fuzzy Sets and Systems. — 1995. — No. 71. — P. 257—264.
59. *Francisco H., Luis M.* Genetic fuzzy systems // Tatra Mount. Math. Publ. — 1997. — No. 13. — P. 93—121.
60. *Chin T. C.* Genetic algorithms for learning the rule base of fuzzy logic controller // Fuzzy Sets and Systems. — 1998. — No. 97. — P. 1—7.
61. *Juang C.-F., Lin J.-Y.* Genetic reinforcement learning through symbiotic evolution for fuzzy controller design // IEEE Trans. Syst., Man and Cybern. Ptb. Cybernetics. — 2000. — Vol. 30, No. 2. — P. 290—302.
62. *Rajapakse A., Furuta K., Kondo A.* Evolutionary learning of fuzzy logic controllers and their adaptation through perpetual evolution // IEEE Trans. Fuzzy Systems. — 2002. — Vol. 10, No. 3. — P. 309—321.
63. *Chen C., Wong C.-C.* Self-generating rule-mapping fuzzy controller design using a genetic algorithm // IEE Proc. Control Theory Appl. — 2002. — Vol. 149, No. 2. — P. 143—148.
64. *Karr C. L.* Genetic algorithms for fuzzy controllers // A I Expert. — 1991. — Vol. 6, No. 1. — P. 26—33.
65. *Karr C. L., Gentry E.* Fuzzy control of PH using genetic algorithms // IEEE Trans. Fuzzy Systems. — 1993. — Vol. 1, No. 1.
66. *Linkens D. A., Nyongesa H. O.* Genetic algorithms for fuzzy control // IEE Proc. Control Theory Appl. — 1995. — Vol. 142, No. 3. — P. 161—185.
67. *Кудинов Ю. И., Халов Е. А., Кудинов И. Ю.* и др. Разработка нечеткой обучаемой системы управления // Промышленные АСУ и контроллеры. − 2004. − № 2. − С. 25−29.
68. *Гнеденко Б. В.* Курс теории вероятностей. — М.: Мир, 1961.
69. *Худсон Д.* Статистика для физиков. − М.: Мир, 1967.
70. *Ротач В. Я.* Экспертная оценка алгоритмов управления методами нечеткой логики и теории вероятностей // Теплоэнергетика. − 2002. − № 4. − С. 51−56.
71. *Pivonka P., Drejl M.* Use of fuzzy PID controllers in fuzzy control of coal power plants // Proceedings of Fuzzy-96. — Zittau, Germany, 1996.
72. *Guilamo P.G.* Fuzzy Control // Measurements and Control. — October 1987.
73. *Ротач В. Я.* О фази-ПИД-регуляторах // Теплоэнергетика. — 1999. — № 8. — С. 32—36.
74. *Ротач В. Я.* Возможен ли синтез нечетких регуляторов с помощью теории нечетких множеств // Промышленные АСУ и контролеры. — 2004. — № 1. — С. 33—34.
75. *Прангишвили И. В., Пащенко Ф. Ф., Бусыгин Б. П.* Системные законы и закономерности в электродинамике, природе и обществе. — М.: Наука, 2001.
76. *Zhang J., Zhang P., Li R.* Analysis and design of fuzzy controller based on fuzzy reaching law // Proceedings of the 14-th World Congress IFAC. — Beijing, 1999. — P. 189—194.
77. *Hu B., Mann G., Gosine R.* How to evaluate fuzzy PID controllers without using process information // Proceedings of the 14-th World Congress IFAC. — Beijing, 1999. — P. 177—182.
78. *Zhang H., Chan C., Chenng K., Ye Y.* Fuzzy art map neural network and its application to fault diagnosis of navigation systems // Automatica. — 2001. — Vol. 37. — P. 1065—1070.

☎ *(0742) 32-80-53*

☎ *(095) 978-95-39*

☎ *(095) 334-90-20*

E-mail: feodor@ipu.ru

❑

УДК621.031:681.332.5

МОНИТОРИНГ НЕСТАЦИОНАРНЫХ ПРОЦЕССОВ: АНАЛИЗ И ИССЛЕДОВАНИЕ ИЗМЕНЕНИЙ СВОЙСТВ СТАЦИОНАРНОСТИ

Е. А. Гребенюк

Институт проблем управления им. В. А. Трапезникова, г. Москва

Рассмотрен алгоритм текущего обнаружения изменений свойств стационарности процесса: изменения состояния процесса от стационарного к нестационарному и наоборот. Дан анализ влияния изменений свойств процесса на коинтеграционные связи.

ВВЕДЕНИЕ

Статистические методы исследования систем, описываемых временными рядами, наряду с традиционными областями приложения, такими как техника, геофизика, геология, сейсмология, медицина, в последнее время стали применяться и для исследования процессов и явлений, протекающих в экономике и обществе. Эти процессы в большинстве своем нестационарные, характеризуются малым числом наблюдений и наличием ряда циклов или этапов развития, на каждом из которых достаточно сильно изменяются закономерности их поведения.

Современные методы анализа систем, описываемых нестационарными процессами, широко оперируют понятием связей между ними. Наличие таких связей обусловлено существованием объективных закономерностей развития анализируемых систем, заключающихся в том, что между изменениями нестационарных временных рядов, описывающих систему, существует долгосрочное равновесие. Например, обменные курсы валют изменяются таким образом, чтобы сохранялся паритет покупательной способности, факторы, вызывающие развитие заболеваний, отражаются на развитии заболеваемости, убытки предприятий конкретного региона увеличивают объемы неплатежей в регионе и пр.

Неизменность связей поддерживает устойчивость системы, их нарушение может существенно изменить поведение процесса, поэтому анализ динамики связей является важным элементом анализа поведения системы, описываемой временными рядами. Для описания связей в классе интегрированных нестационарных процессов введено понятие коинтеграции [1].

Исследование и мониторинг поведения системы предполагает наличие ее адекватной модели. В системе, описываемой временными рядами, модель можно получить путем обработки результатов наблюдений за ее поведением. Для анализа временных рядов широко применяются параметрические модели. Методам построения параметрических моделей и проверки наличия коинтеграционных связей в случае отсутствия изменений в интервале наблюдения посвящено большое число работ (см., например, [2—7]).

Однако, параметры модели могут скачкообразно изменять свои значения, реагируя на внешние события. Пример такого события — дефолт в августе 1998 г., резко изменивший поведение индексов в российской экономике. Наличие изменений в процессах снижает эффективность алгоритмов, построенных без учета этих изменений. В работах [8—11] описываются методы, учитывающие наличие изменений в моделях. Однако они предназначены для построения модели со структурными изменениями после окончания процесса наблюдения за системой. Такая модель может быть использована для прогноза только до момента возникновения новых изменений в процессе.

Если после построения модели в системе произошли изменения, то при прогнозировании будущих значений процесса возникают ошибки. По-

этому после получения адекватной модели необходимо проверять, сохраняет ли она адекватность. Проверка адекватности модели в режиме получения текущих наблюдений и своевременное выявление отклонений позволяет прогнозировать развитие системы и предотвращать возможные кризисные и неблагоприятные ситуации. Цель настоящей статьи состоит в формировании методов, предназначенных для мониторинга нестационарного процесса с изменением свойств и основанных на алгоритмах обнаружения отклонений системы от модели в режиме получения текущих наблюдений.

ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ И ПОСТАНОВКА ЗАДАЧИ

Определение 1. *Случайный процесс y_t называется интегрированным порядка d, если применение к нему d раз разностного оператора превращает его в стационарный, и обозначается I(d).*

Стационарный процесс является интегрированным нулевого порядка и обозначается $I(0)$.

Определение 2. *Если линейная комбинация случайных процессов интегрированных порядка d, является интегрированным процессом порядка $d - 1$, то эти процессы называются коинтегрированными, а вектор коэффициентов линейной комбинации — коинтегрирующим вектором.*

Постановка задачи. Рассмотрим систему, поведение которой описывается случайной векторной последовательностью временных рядов:

$$y_t = \sum_{j=1}^{p} \Pi_i y_{t-i} + \varepsilon_t,$$

где y_t — вектор размерности $n \times 1$, каждая компонента которого представляет собой процесс типа $I(0)$ либо $I(1)$, ε_t — гауссовский белый шум, $\Pi(L) = I - \sum_{i=1}^{p} \Pi_i L^i$ — матричный полином, имеющий $k < n$ единичных корней, L — оператор запаздывания. Стандартными преобразованиями эта модель преобразуется к виду [7]:

$$\Delta y_{1,t} = \mu_{1,t}, \qquad (1a)$$

$$\Delta y_{1,t} - \beta y_{1,t} = \mu_{2,t} \qquad (1б)$$

где $y_t = (y_{1,t}, y_{2,t})'$, $y_{1,t}$ — вектор компонент размерности $k \times 1$, $y_{2,t}$ — вектор компонент размерности $r \times 1$, $n = r + k$, $\mu_t = (\mu_{1,t}, \mu_{2,t})' = D(L)\varepsilon_t$, матрица $D(1)$ имеет полный ранг. Уравнения (1*б*) описывают коинтеграционные связи между компонентами $y_{1,t}$ и $y_{2,t}$. Предположим, что в компонентах $y_{1,t}$ в неизвестные моменты времени $t = t_a$ возникают изменения следующих типов:

- изменение стохастического тренда:

$$\Delta y_{1,t} = \begin{cases} \mu^0_{1,t}, \text{ если } t \le t_a \\ \mu^1_{1,t}, \text{ если } t > t_a, \end{cases} \qquad (2)$$

где $\mathrm{var}(\mu^0_{1,t}) \ne \mathrm{var}(\mu^1_{2,t})$; $\mathrm{var}(\cdot)$ — дисперсия величины $(\cdot)$;

- изменение свойств стационарности процесса:

$$\Delta y_{1,t} = \begin{cases} \mu^0_{1,t}, \text{ если } t \le t_a \\ -\alpha y_{t-1} + \mu^0_{1,t}, \text{ если } t_b > t > t_a \\ \mu^1_{1,t}, \text{ если } t \ge t_b, \end{cases} \qquad (3)$$

где $0 < \alpha < 1$, $\mu^0_{1,t}$, $\mu^1_{1,t}$ — стационарные процессы.

Требуется построить алгоритмы обнаружения отклонений системы от модели, работающие в режиме текущего получения наблюдений.

Обсуждение. В рассматриваемой постановке первые k уравнений преобразованной системы описывают независимо изменяющиеся компоненты, последние r уравнений описывают коинтеграционные связи между ее компонентами. Методы построения моделей вида (1*а*), (1*б*) с изменениями свойств типа (2) и (3) описаны во многих работах [2—7]. Для разработки алгоритмов обнаружения отклонений системы от модели в режиме текущего получения наблюдений решаются:

— задача анализа влияния изменений (2) и (3) на коинтеграционные связи (1*б*);

— задача текущего обнаружения изменений типа (2) и (3).

Задача текущего обнаружения изменений типа (2) решается применением алгоритмов, рассмотренных в работах [12, 13]. Для обнаружения изменения свойств стационарности процесса применяется алгоритм, описанный в настоящей статье.

Анализ влияния изменений свойств процесса на коинтеграционные связи

Рассмотрим изменения коинтеграционных связей при изменениях свойств процесса (2) и (3) для модели (1*а*), (1*б*). При анализе процессов $y_{i,t}$, $i = 1, 2$, до изменения свойств и после будем пользоваться обозначением:

$$\Delta y_{1,t} = \begin{cases} y^0_{1,t}, \text{ если } t \le t_a \\ y^1_{1,t}, \text{ если } t > t_a, \end{cases}$$

где $i = 1, 2$.

Изменение стохастического тренда. Перепишем соотношение (2) в виде

$$\Delta y_{1,\,t} = \begin{cases} y^0_{1,\,t-1} + \mu^0_{1,\,t}, \text{ если } t \le t_a \\ y^0_{1,\,t-1} + \mu^1_{1,\,t}, \text{ если } t = t_a + 1 \\ y^1_{1,\,t-1} + \mu^1_{1,\,t}, \text{ если } t \ge t_a + 1, \end{cases}$$

и рассмотрим две возможные ситуации:

- процесс $y_{2,\,t}$ изменяется в соответствии с моделью (1*б*): $y_{2,t} = \beta y_{1,t} + \mu_{2,t}$, тогда коинтеграционное соотношение сохраняется, но в силу условия (2) изменяется дисперсия разностей процесса $y_{2,t}$: $\Delta y_{2,t} = \beta y_{1,t}$;
- процесс $y_{2,t}$ после изменения $y_{1,t}$ продолжает по-прежнему изменяться по закону: $y^1_{2,\,t}$, $t = \beta y^0_{1,\,t} + \mu_{2,t}$. В этом случае дисперсия разностей процесса $y_{2,\,t}$ не изменяется после момента t_a, но коинтеграция нарушается. Разность

$$y_{2,\,t} - \beta y_{1,\,t} \qquad (4)$$

после изменения принимает вид:

$$y^1_{2,\,t} - \beta y^0_{1,\,t-1} = \beta(y^0_{1,\,t} - y^1_{1,\,t}) + \mu_{2,\,t} =$$
$$= \beta(y^0_{1,\,t-1} + \mu^0_{1,\,t} - y^0_{1,\,t-1} - \mu^1_{1,\,t}) + \mu_{2,\,t} =$$
$$= \beta(\mu^0_{1,\,t} - \mu^1_{1,\,t}) + \mu_{2,\,t},$$

если $t = t_a + 1$,

$$y^1_{2,\,t} - \beta y^1_{1,\,t-1} = \beta(y^0_{1,\,t} - y^1_{1,\,t}) + \mu_{2,\,t} =$$
$$= \beta(y^0_{1,\,t-1} + \mu^0_{1,\,t} - y^1_{1,\,t-1} - \mu^1_{1,\,t}) + \mu_{2,\,t} =$$
$$= k\beta(\mu^0_{1,\,t} - \mu^1_{1,\,t}) + \mu_{2,\,t},$$

если $t = t_a + k$,

т. е. дисперсия разности (4) растет, и процесс $y_{2,\,t} - \beta y_{1,\,t}$ становится нестационарным.

Изменение характера стационарности процесса $y_{1,t}$. Как и в случае изменения стохастического тренда, рассмотрим две возможные ситуации:

- процесс $y_{2,\,t}$ после изменения свойств процесса $y_{1,\,t}$ изменяется в соответствии с моделью (1*б*): $y_{2,\,t} = \beta y_{1,\,t} + \mu_{2,\,t}$ тогда процессы $y_{1,\,t}$ и $y_{2,\,t}$ становятся стационарными:

$$y^1_{1,\,t} = (1 - \alpha)y^0_{1,\,t-1} + \mu^0_{1,\,t}, \text{ если } t = t_a + 1,$$
$$y^1_{2,\,t} = \beta(1 - \alpha)y^0_{1,\,t-1} + \beta\mu^0_{1,\,t} + \mu_{2,\,t}, \text{ если } t = t_a + 1,$$
$$y^1_{1,\,t} = (1 - \alpha)y^1_{1,\,t-1} + \mu^0_{1,\,t}, \text{ если } t > t_a + 1,$$
$$y^1_{2,\,t} = \beta(1 - \alpha)y^1_{1,\,t-1} + \beta\mu^0_{1,\,t} + \mu_{2,\,t}, \text{ если } t > t_a + 1.$$

- процесс $y_{2,\,t}$ после изменения свойств процесса $y_{1,\,t}$ продолжает по-прежнему изменяться по закону: $y^1_{2,\,t} = \beta y^1_{1,\,t} + \mu_{2,\,t}$. В этом случае процесс $y_{2,\,t}$ остается нестационарным:

$$y^1_{1,\,t} = \beta(y^0_{1,\,0} + \sum_{j=1}^{t} \mu^0_{1,\,t}) + \mu_{0,\,t},$$

а коинтеграционные связи нарушаются. Разность (4) после момента $t = t_a$ изменения свойств принимает вид:

$$y^1_{2,\,t} - \beta y^0_{1,\,t} = \beta(y^0_{1,\,t} - y^1_{1,\,t}) + \mu_{2,\,t},$$

так как $y^0_{1,\,t} - y^1_{1,\,t}$ представляет собой разность стационарного и нестационарного процессов, то

Диагностика изменений двух коинтегрированных процессов

Изменения процесса $y_{1,t}$	Изменения процесса $y_{2,t}$	Изменения связи $y_{2,t} - \beta y_{1,t}$	Признаки изменений
Изменение стохастического тренда, процесс остается нестационарным	Изменение стохастического тренда, процесс остается нестационарным	Коинтеграционная связь сохраняется	Изменение стохастического тренда процессов $y_{1,t}$ и $y_{2,t}$, процесс $y_{2,t} - \beta y_{1,t}$ остается стационарным
Изменение стохастического тренда, процесс остается нестационарным	Без изменений	Нарушение коинтеграционной связи	Изменение стохастического тренда процесса $y_{1,t}$, переход процесса $y_{2,t} - \beta y_{1,t}$ от стационарного состояния к нестационарному
Переход процесса в стационарное состояние	Переход процесса в стационарное состояние	Коинтеграционная связь теряет смысл	Переход процессов $y_{1,t}$ и $y_{2,t}$ от нестационарного состояния к стационарному
Переход процесса в стационарное состояние	Без изменений	Нарушение коинтеграционной связи	Переход процесса $y_{1,t}$ от нестационарного состояния к стационарному, переход процесса $y_{2,t} - \beta y_{1,t}$ от стационарного состояния к нестационарному

разность $y^1_{2,t} - \beta y^0_{1,t}$ — нестационарная, следовательно, ковариационные связи нарушаются.

Изменения в системе могут быть диагностированы по некоторому набору признаков. Перечень изменений и набор признаков, по которым их можно диагностировать, представлены в таблице.

Алгоритм обнаружения изменения стохастического тренда рассматривался в работах [12, 13]. Алгоритмы обнаружения перехода процесса от стационарного состояния к нестационарному по текущим наблюдениям и наоборот рассмотрены далее.

Алгоритм текущего обнаружения изменения свойств нестационарности процесса (нестационарный, стационарный)

Построим алгоритм обнаружения изменений по текущим наблюдениям. Мы будем здесь рассматривать покомпонентное обнаружение изменений. Пусть каждая компонента процесса $y_{1,t}$ описывается моделью $\Delta y_t = \mu_t$, где

$$\mu_t = \sum_{j=0}^{k} \alpha_j \varepsilon_{t-j}, \qquad (5)$$

ε_t — гауссовский белый шум.

В соответствии с выражением (3) в некоторый момент времени $t = t_a$ в процессе происходит изменение свойства стационарности:

$$\Delta y_1 = \begin{cases} \mu_t, \text{ если } t \le t_a \\ -\alpha y_{t-1} + \mu_t, \text{ если } t > t_a, \end{cases} \qquad (6)$$

где $0 < \alpha < 1$, и в некоторый момент времени $t = t_b > t_a$ в стационарном процессе y_t возникают изменения:

$$\Delta y_1 = \begin{cases} -\alpha y_{t-1} + \mu_t, \text{ если } t_a \le t \le t_b \\ \mu_t, \text{ если } t > t_b, \end{cases} \qquad (6a)$$

Как известно, дисперсия нестационарного процесса неограниченно возрастает с увеличением времени наблюдений: $\mathrm{var}(y_t) \to \infty$ при $t \to \infty$. Оценим дисперсию процессов y_t и Δy_t после того, как он становится стационарным.

Теорема 1. *Если в процессе происходят изменения свойств* (6)*, то:*

1) *дисперсия процесса y_t после изменения свойств ограничена сверху величиной*

$$P_1 = \mathrm{var}(\mu_t)\left(\frac{1}{1-(1-\alpha)} + \frac{1}{1-(1-\alpha)^2} + \ldots + \frac{1}{1-(1-\alpha)^k}\right),$$

где $\mathrm{var}(\mu_t)$ — *дисперсия процесса* μ_t*;*

2) *дисперсия процесса Δy_t стремится к величине*

$$P_2 = \alpha\left(\frac{\sum_{i=1}^{k}\alpha_i^2}{1-(1-\alpha)^2} + \frac{\sum_{i=1}^{k-1}\alpha_i\alpha_{i+1}}{1-(1-\alpha)} + \frac{\sum_{i=1}^{k-2}\alpha_i\alpha_{i-2}}{1-(1-\alpha)} + \ldots + \frac{\alpha_k\alpha_1}{1-(1-\alpha)^k}\right)\mathrm{var}(\varepsilon).$$

Если в процессе происходят изменения свойств (6*a*)*, то дисперсия процесса Δy_t стремится к величине*

$$P_3 = \sum_{i=1}^{k}\alpha_i^k\,\mathrm{var}(\varepsilon) = \frac{1-\alpha^{k+1}}{1-\alpha}\,\mathrm{var}(\varepsilon),$$

k — порядок модели (5).

Рассмотрим статистику $T = \Delta y_t' \Delta y_{t-k}$, представляющую собой функцию от одинаково распределенных независимых случайных величин ε_t, ε_{t-1}, ..., ε_0, где $t \to \infty$, и исследуем ее свойства.

Теорема 2. *Если в процессе происходят изменения свойств, то:*

1) *математическое ожидание статистики T после изменения свойств* (6) *ограничено снизу величиной:*

$$M = -\left(\frac{1-\alpha}{1+\alpha}\right)\mathrm{var}(\mu_t);$$

2) *математическое ожидание статистики T после изменения свойств* (6*a*) *равно нулю, k — порядок модели* (5).

(Доказательства теорем 1 и 2 основаны на вычислении соответствующих статистик процессов y_t и Δy_t до момента изменения t_a и после).

Алгоритм обнаружения изменения свойств (6) представляет собой алгоритм кумулятивных сумм для текущего обнаружения одновременного изменения следующих параметров:

— уменьшения дисперсии процесса y_t до величины P_1;

— увеличения дисперсии процесса Δy_t до величины P_2;

— уменьшения математического ожидания статистики T до величины M.

Алгоритм обнаружения изменения свойств (6*a*) представляет собой алгоритм кумулятивных сумм для текущего обнаружения одновременного изменения следующих параметров:

— увеличения дисперсии процесса y_t больше величины P_1;

— уменьшения дисперсии процесса Δy_t до величины P_3;

— увеличения математического ожидания статистики T до нуля.

Алгоритм кумулятивных сумм в случае точно известных параметров процесса до и после изменения свойств предложен в работе [14], в случае заданных границ параметров до и после изменения свойств — в работах [12, 13].

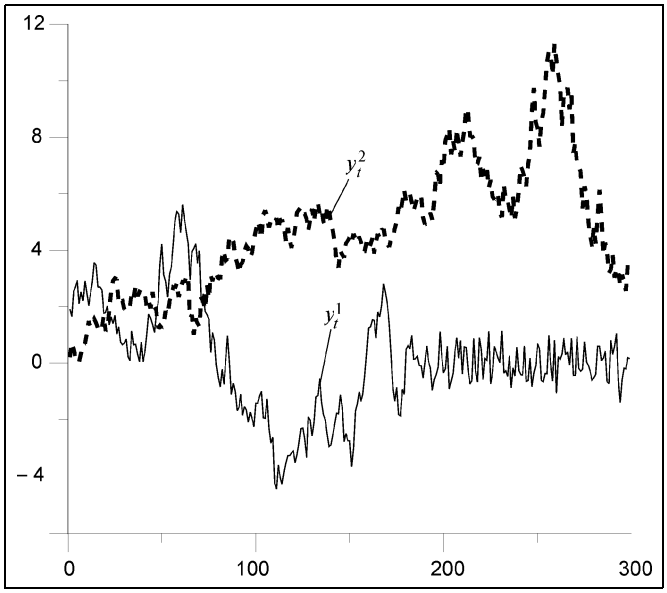


Рис. 1. Графики процессов y_t^1 и y_t^2

Применение алгоритма обнаружения нестационарности для анализа изменения биржевых индексов

Рассмотрим пример, иллюстрирующий работу алгоритма. На рис. 1 приведены графики двух смоделированных рядов

$$y_t^1 = \begin{cases} y_{t-1}^1 + \varepsilon_t, \text{ если } t \le 200 \\ -0{,}8\, y_{t-1}^1 + \varepsilon_t, \text{ если } t > 200 \end{cases} \quad (7)$$

и

$$y_t^2 = \begin{cases} y_{t-1}^2 + \varepsilon_t^1, \text{ если } t \le 200 \\ y_{t-1}^2 + \varepsilon_t^2, \text{ если } t > 200 \end{cases} \quad (8)$$

где ε_t, ε_t^1, ε_t^2 — гауссовский белый шум, $\mathrm{var}(\varepsilon_t) = \mathrm{var}(\varepsilon_t^1) = 0{,}7$, $\mathrm{var}(\varepsilon_t^2) = 1$. Как следует из моделей (7) и (8), первый ряд после 200-й точки становится стационарным, а во втором происходит изменение стохастического тренда. Алгоритм обнаружения настраивается по первым 100 точкам. Для каждой из трех решающих функций определяются пороги, превышение решающей функцией порога дает сигнал об изменении свойств. Обнаружение перехода процесса из нестационарного состояния в стационарное фиксируется, когда все решающие функции будут находиться выше порога. Обнаружение изменения стохастического тренда фиксируется, когда обнаруживается изменение дисперсии процесса Δy_t. На рис. 2 и 3 показаны графики процессов y_t^1 и y_t^2 и решающие функции алгоритма обнаружения изменения свойств стационарно-

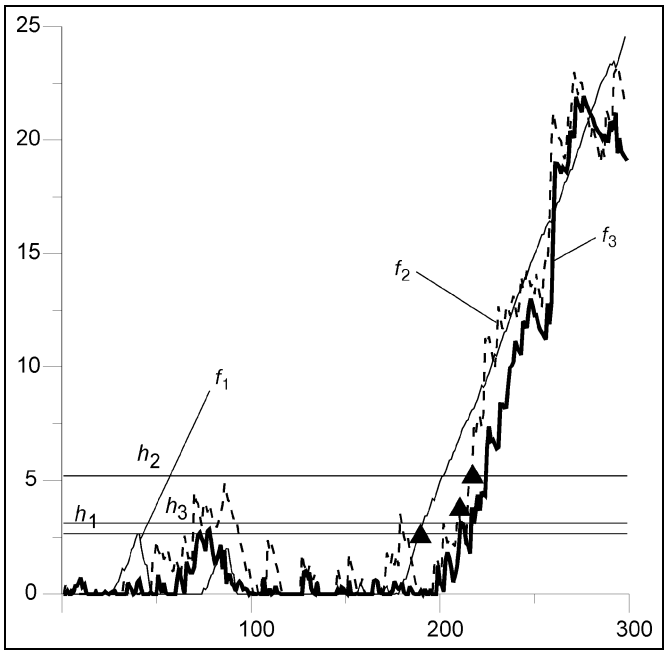


Рис. 2. Решающие функции алгоритма обнаружения изменения нестационарности f_i, i = 1, 2, 3:

▲ — момент превышения решающей функцией порога; h_i — порог решающей функции

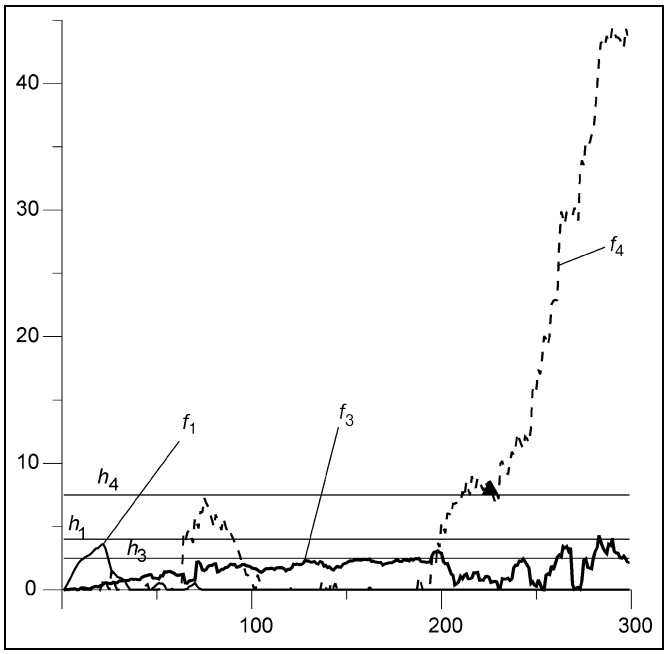


Рис. 3. Решающие функции алгоритма обнаружения изменения стохастического тренда f_i, i = 1, 3, 4:

▲ — момент превышения решающей функцией порога; h_i — порог решающей функции

сти процесса, где f_1 — решающая функция алгоритма обнаружения (примененного к процессу y_t) уменьшения дисперсии; f_2 — решающая функция алгоритма обнаружения (примененного к процессу Δy_t) увеличения дисперсии; f_3 — решающая функция алгоритма обнаружения (примененного к значениям статистики T) уменьшения математического ожидания; f_4 — решающая функция алгоритма обнаружения (примененного к процессу Δy_t) изменения дисперсии процесса.

Как видно из рисунков, в случае изменения стохастического тренда происходит обнаружение дисперсии процесса Δy_t^2, дисперсия процесса y_t^2 и математическое ожидание статистики T не уменьшаются. В случае изменения процесса от нестационарного к стационарному происходит уменьшение дисперсии процесса y_t^1, увеличение дисперсии процесса Δy_t^1, увеличение математического ожидания статистики T до нуля.

ЛИТЕРАТУРА

1. *Engle R. F., Granger C. W. J.* Cointegration and Error Correction: Representation, Estimation and Testing // Econometrica. — 1987. — Vol. 55. — P. 251—276.
2. *Fuller W. A.* Introduction to Statistical Time Series. N.-Y.: Willey, 1996.
3. *Hamilton J. D.* Time Series Analysis. Princeton University Press, 1994. — P. 296.
4. *Granger C. W. J.* Some Properties of Time Series Data and Their Use in Econometric Model Specification // Journal of Econometrics. — 1981. — Vol. 16. — P. 121—130.
5. *Sims C. A., Stock J. H., Watson M. W.* Inference in Linear Time Series Models with Some Unit Roots // Econometrica. — 1990. — Vol. 58. — P. 113—144.
6. *Stock J. H.* Asymptotic Properties of Least Squares Estimators of Cointegrating Vectors // Econometrica. — 1987. — Vol. 55. — P. 1035—1056.
7. *Watson M. W.* Vector Avtoregression and Cointegration // Handbook of Econometrics. Amsterdam, 1994. — Vol. 4. — P. 2844—2915.
8. *Bai J.* Estimating Multiple Breaks One at a Time // Econometric Theory. — 1997. — Vol. 13. — P. 315—52.
9. *Bai J. and Perron P.* Estimating and Testing Linear Models with Multiple Structural Changes // Econometrica. — 1998. — Vol. 66. — P. 47—78.
10. *Perron P.* The Great Crash, the Oil Price Shock, and the Unit Root Hypothesis // Econometrica. — 1989. — Vol. 57. — P. 1361—401.
11. *Zivot E. and Andrews D. W. K.* Further Evidence on the Great Crash, the Oil Price Shock? And the Unit Root Hypothesis // Journal of Business and Economic Statistics. — 1992. — No. 10. — P. 251—70.
12. *Гребенюк Е. А.* Обнаружение изменений свойств нестационарных случайных процессов // Автоматика и телемеханика. — 2003. № 12. — С. 25—41.
13. *Гребенюк Е. А.* Анализ и оперативная диагностика систем, описываемых нестационарными случайными процессами // Проблемы управления. — 2003. № 4. — С. 23—29.
14. *Page E. S.* Continuous inspection schemes // Biometrika. — 1954. — Vol. 41. — P. 100—115.

☎ *(095) 334-76-40*

E-mail: grebenuk@yahoo.com ❑

Уважаемые читатели!

Посетите Интернет-сайт теории управления организационными системами www.mtas.ru

Цель сайта — предоставление специалистам по теории и практике управления организационными системами (ученым, преподавателям, аспирантам, студентам, а также реальным управленцам) доступа к ресурсам, отражающим современное состояние теории и возможности обмена идеями и результатами.

На сайте представлены разделы:

Теория — с обзором теории управления организационными системами, глоссарием, информацией для аспирантов;

Практика — с обзором результатов внедрения механизмов управления в реальных организациях;

Библиография — около 2000 публикаций по теории управления, снабжена классификатором и аннотациями;

Электронная библиотека — более 100 полнотекстовых монографий, статей и учебных пособий;

а также многое другое.

На сайте работает **форум**, где можно обсудить вопросы, относящиеся к математике, экономике, управлению организациями, узнать новости теории управления и ознакомиться с тематикой, временем и местом проведения планируемых конференций и семинаров.

С октября 2003 г. сайт посетили более 10 000 человек.

УДК 330.4;681.513.4

МНОГОКРИТЕРИАЛЬНАЯ ОПТИМИЗАЦИЯ ПОРТФЕЛЯ ИНВЕСТИЦИЙ. ДЕТЕРМИНИРОВАННЫЙ СЛУЧАЙ

А. С. Рыков, Р. Р. Исходжанов

Московский государственный институт стали и сплавов (технологический университет)

Предложен метод решения многокритериальной задачи условной оптимизации распределения ресурсов при изменении состава портфеля инвестиций, относящийся к классу методов деформируемых конфигураций. Рассмотрены вопросы его практического применения.

ВВЕДЕНИЕ

Определение оптимального состава портфеля инвестиций — одна из важнейших задач деятельности как профессиональных игроков фондового рынка, так и владельцев или распорядителей капитала, не задействованного в текущий момент времени в сфере производства товаров и услуг. Традиционные методики ее решения детально разработаны и исследованы, однако отражают только один аспект выгоды инвестора — доход от реализации портфеля инвестиций — и предполагают множество допущений.

Многокритериальная модель портфеля инвестиций, позволяющая учесть разнородные цели инвестора, а также влияние факторов внешней среды путем агрегирования частных критериев качества портфеля в единую целевую функцию и сценарного подхода к его моделированию, предложена в работе [1]. Данная модель позволяет оценить качество портфеля заданного состава, однако не позволяет определить его оптимальный состав, наилучшим образом соответствующий целям данного инвестора-ЛПР (лица, принимающего решения) в текущей ситуации в текущий момент времени.

Для эффективного решения задачи оптимизации портфеля инвестиций требуется применение специальных методов оптимизации, опирающихся на современный математический аппарат и эвристические приемы, обусловленные ее характером.

В статье рассматривается модель портфеля инвестиций, позволяющая сформулировать постановку многокритериальной задачи условной оптимизации портфеля инвестиций, предложен метод решения задачи оптимизации из класса методов деформируемых конфигураций [2, 3]. Решение задачи показывает состав портфеля инвестиций, реализация которого позволит в наибольшей степени достичь разнообразных целей инвестора-ЛПР.

ПОСТАНОВКА ЗАДАЧИ ОПТИМИЗАЦИИ

Задача оптимизации портфеля инвестиций решается в детерминированном случае — при условии, что все значения аргументов и критериев выражены действительными числами.

Приведем в обобщенном виде постановку задачи оптимизации портфеля инвестиций, подробно описанной в работе [1], а затем укажем способы решения поставленной задачи. Пусть даны:

— исходное множество допустимых решений X — подмножество $(n + m)$-мерного пространства аргументов V^{n+m}:

$$X = \{v = (v_1, \ldots, v_{n+m}) \in V^{n+m} | h_i(v) = 0, \; i \in I_Y \cup I_Z \cup I_G, \; g_j(v) \le 0, \; j \in J_Y \cup J_Z \cup J_G\}, \quad (1)$$

где $h_i(v)$, $g_j(v)$ − ограничения, I_Y, J_Y — множества индексов ограничений проектов, I_Z, J_Z — множества индексов ограничений ресурсов, I_G, J_G — множества индексов ограничений целей, v_i, $i = 1, \ldots, n - 1$ — объем вложения в i-й инвестиционный проект, v_n — объем инвестирования в ликвидный безрисковый актив, v_i, $i = n + 1, \ldots, n + m$ — объем использования доступных источников ресурсов;

— отображение точки исходного множества X в точку пространства частных критериев качества $C^{\bar{u}}$:

$$U: V^{n+m} \to C^{\bar{u}}, \quad (2)$$

где $\bar{u}$ — число частных критериев качества (размерность критериального пространства);

— функция полезности, определенная на пространстве частных критериев $C^{\bar{u}}$ (в явном виде либо в виде неформализованных предпочтений инвестора-ЛПР):

$$U^*: C^{\bar{u}} \to \mathrm{R}. \qquad (3)$$

Пусть функция полезности выбрана так, что для $\forall c_1, c_2 \in C^{\bar{u}}$ выполняются соотношения:

$$\begin{cases} U^*(c_1) < U^*(c_2) \Rightarrow c_1 \prec c_2 \\ U^*(c_1) = U^*(c_2) \Rightarrow c_1 \propto c_2 \\ U^*(c_1) > U^*(c_2) \Rightarrow c_1 \succ c_2. \end{cases} \qquad (4)$$

Рассмотрим композицию преобразований U и U^*:

$$F = U \times U^*.$$

Тогда

$$F: V^{n+m} \to R.$$

Очевидно, что соотношения (4) сохранятся для $\forall v_1, v_2 \in V^{n+m}$:

$$\begin{cases} F(v_1) < F(v_2) \Rightarrow v_1 \prec v_2 \\ F(v_1) = F(v_2) \Rightarrow v_1 \propto v_2 \\ F(v_1) > F(v_2) \Rightarrow v_1 \succ v_2. \end{cases}$$

Постановка задачи условной оптимизации в этом случае будет иметь следующий вид:

$$v^* = \arg\min_{v \in X} (-F(v)). \qquad (5)$$

Отметим, что мы не налагаем ограничений на определенность целевой функции $F(v)$ на всем пространстве аргументов V^{n+m}, поэтому за пределами исходного множества допустимых решений X вычисление значений целевой функции (функции полезности) в ходе решения задачи оптимизации производиться не будет.

МЕТОД РЕШЕНИЯ ЗАДАЧИ ОПТИМИЗАЦИИ

Предлагаемая процедура решения задачи условной оптимизации (1)—(5) разбивается на следующие этапы.

1. Снижение размерности решаемой задачи с $n + m$ до $n - 1$ и формирование в подпространстве V^{n-1} проекции X^{n-1} множества допустимых значений X.

2. Определение числа начальных точек $\bar{i}$.

3. Генерация множества V_0, состоящего из $\bar{i}$ начальных точек в пространстве переменных V^{n-1}:

$$V_0 = \{v_{0,i} | v_{0,i} \in V^{n-1}, i = 1, ..., \bar{i}\}.$$

4. Поиск для каждой начальной точки из множества V_0 допустимой начальной точки в исходном множестве X^{n-1}:

$$\tilde{V}_0 = \{\tilde{v}_{0,i} | \tilde{v}_{0,1} \in X^{n-1}, i = 1, ..., \bar{i}\}.$$

5. Поиск решения задачи (5) для каждой допустимой начальной точки $\tilde{v}_{0,i}$, $i = 1, ..., \bar{i}$, формирование множества решений:

$$\tilde{V}_{\mathrm{opt}} = \{\tilde{v}_{\mathrm{opt},i} | \tilde{v}_{\mathrm{opt},i} \arg \max_{v \in X^{n-1}} F(Q^{-1}(v)), i = 1, ..., \bar{i}\}.$$

1. Ранжирование решений $\tilde{v}_{\mathrm{opt},i}$, $i = 1, ..., \bar{i}$ по значению целевой функции $F(Q^{-1}(v))$ либо выбор в диалоге с инвестором-ЛПР лучшего решения:

$$\tilde{x} = \arg\max_{v \in \tilde{V}_{\mathrm{opt}}} F(Q^{-1}(x)).$$

Прокомментируем кратко представленную процедуру решения задачи оптимизации. На первом этапе из-за учета содержательных особенностей модели снижается размерность пространства, в котором ищется решение задачи оптимизации.

Предполагается, что целевая функция $F(v)$ может в общем случае иметь сложный для оптимизации характер, например, может быть многоэкстремальной и овражной. Для преодоления отмеченных сложностей предлагается решать множество задач оптимизации (5), запуская итеративный метод решения из разных начальных условий. Поэтому на втором этапе предложено выбрать число начальных точек $\bar{i}$, а на третьем этапе сформировать множество начальных точек V_0.

Начальные точки из множества V_0 могут не принадлежать множеству допустимых решений X. На четвертом этапе с помощью решения вспомогательных оптимизационных задач по минимизации отклонения точки от ограничений, образующих множество X, осуществляется поиск для каждой исходной точки допустимого решения. В результате получаем множество $\bar{i}$ допустимых начальных точек $\tilde{V}_0$.

На пятом этапе решается $\bar{i}$ задач оптимизации (5) и формируется множество решений $\tilde{V}_{\mathrm{opt}}$.

На последнем этапе полученные решения либо ранжируются, либо в диалоге с инвестором-ЛПР определяется лучшее решение.

Такова общая схема решения задачи оптимизации. При описании процедуры не обсуждался важнейший вопрос: каким методом решать возникающие оптимизационные задачи?

Задача (1)—(5) относится к классу задач условной нелинейной конечномерной оптимизации.

Заметим, что в общем случае производные целевой функции и ограничений могут быть неизвестны. Для решения задачи оптимизации были выбраны методы из класса методов деформируемых конфигураций, хорошо зарекомендовавшие себя при оптимизации нелинейных, недифференцируемых функций, в том числе овражного типа, в задачах оптимизации средней размерности [2–4].

Перейдем к более подробному описанию отдельных элементов представленной процедуры решения задачи условной оптимизации.

Снижение размерности задачи оптимизации

Рассмотрим способ снижения размерности задачи оптимизации путем учета специфики рассматриваемой модели портфеля инвестиций.

В силу специфики модели портфеля инвестиций величины v_i, $i = n, ..., n + m$, функционально зависят от величин v_i, $i = 1, ..., n - 1$. Для рассматриваемой модели эффективным будет наличие не более одной точки с данной комбинацией значений v_i, $i = 1, ..., n - 1$. Отсюда следует, что при решении задачи оптимизации для снижения ее размерности и, следовательно, снижения вычислительной сложности можно воспользоваться преобразованием Q: $V^{n+m} \to V^{n-1}$, что тождественно $Q(v) = \tilde{v}$, $\tilde{v} = \{\tilde{v}_1, ..., \tilde{v}_{n-1}\}$, $v_i = \tilde{v}_i$, $i = 1, ..., n - 1$.

Обратное преобразование будет иметь вид Q^{-1}: $V^{n-1} \to V^{n+m}$, что тождественно $Q^{-1}(\tilde{v}) = v$.

Обозначим отображение исходного множества допустимых решений X в подпространство размерности $n - 1$ преобразованием Q: $X \to X^{n-1}$, и обратное преобразование Q^{-1}: $X^{n-1} \to X$.

Снижение размерности исходного пространства с $n + m$ до $n - 1$ облегчает решение исходной задачи оптимизации.

Поиск допустимых начальных точек

Начальной точкой является исходный состав портфеля инвестиций $v_{0,i} \in V_0$. Сконструируем штрафную функцию

$$P(Q^{-1}(v)) = \\ = \sum_i |h_i(Q^{-1}(v))| - \sum_j \max\{g_j(Q^{-1}(v)), 0\}, \qquad (6)$$

где $i \in I_Y \cup I_Z \cup I_G$, $j \in J_Y \cup J_Z \cup J_G$.

Тогда задача оптимизации штрафной функции будет иметь вид:

$$\text{найти } \tilde{v}_{0,i} = \arg \min_{v \in V^{n-1}} P(Q^{-1}(v)).$$

Для решения этой задачи применялись методы класса деформируемых конфигураций. Ее решением будет первая точка $\tilde{v}_{0,i} \in V^{n-1}$, для которой $P(Q^{-1}(\tilde{v}_{0,i})) = 0$. Очевидно, что $\tilde{v}_{0,i} \in X^{n-1}$.

Расчет значения целевой функции

Применение преобразования Q для снижения размерности задачи позволяет разбить процедуру расчета значения целевой функции $F(Q^{-1}(\tilde{v}))$ в процессе оптимизации на множестве допустимых значений на следующие этапы:

- определение объема инвестирования в доступные инвестиционные проекты по правилам генерации новой альтернативы: $\tilde{v} = \{\tilde{v}_1, ..., \tilde{v}_{n-1}\}$;
- определение оптимального объема использования источников ресурсов при заданном объеме инвестирования с учетом использования ликвидного безрискового актива для компенсации оттоков будущих периодов:

$$v = \{v_1, ..., v_{n+m}\},$$

$$v_i = \begin{cases} \tilde{v}_i, & i = 1, ..., n-1 \\ \arg \min\limits_{x \in X} \sum\limits_{j=1}^{m} z_j(v), & i = n, ..., n+m, \end{cases}$$

$$z_j(v) = \sum_{t=1}^{\bar{t}} \frac{\left|IC_j^t\right|}{\prod\limits_{a=0}^{t} (1 + i_j^a)^{t_a}},$$

$$\begin{cases} v_n = \sum\limits_{j=1}^{n-1} \sum\limits_{t=1}^{\bar{t}} \dfrac{\left|IC_j^t\right|}{\prod\limits_{a=0}^{t} (1 + i_n^a)^{t_a}} \\ \sum\limits_{i=1}^{n} v_i = \sum\limits_{j=n+1}^{n+m} v_j, \end{cases}$$

где IC_j^t — денежный отток j-го инвестиционного проекта в t-м периоде; t — количество периодов в горизонте расчета; i_n^a — ставка дисконтирования ликвидного безрискового актива в a-м периоде; t_a — длительность a-го периода;

- моделирование инвестиционного портфеля заданного состава и расчет частных критериев качества портфеля: $\{U_i(x)\}$, $i = 1, ..., \bar{u}$;
- определение полезности портфеля (значений целевой функции в зависимости от состава портфеля инвестиций):

$$F(x) = U^*[U(v)].$$

Оптимизация на множестве допустимых значений

Начальной точкой для построения траектории оптимизации на множестве допустимых значений является точка $\tilde{v}_{0,i} \in X^{n-1}$. Сконструируем целевую оптимизируемую функцию, которая:

а) позволит отказаться от вычисления значения целевой функции $F(Q^{-1}(v))$ за пределами множества X^{n-1};

б) будет выпуклой при условии выпуклости исходной целевой функции $F(Q^{-1}(x))$ и ограничений задачи (1)—(5).

В качестве штрафной функции будем пользоваться функцией (6). Тогда следующая конструкция обеспечивает выполнение условий а) и б):

$$\tilde{F}(v) = \begin{cases} -F(Q^{-1}(v)), & v \in X^{n-1} \\ -F(Q^{-1}(v)) + P(Q^{-1}(v)), & v \notin X^{n-1}. \end{cases} \quad (7)$$

Очевидно, что решение задачи $\tilde{v}^* = \arg \min\limits_{v \in X^{n-1}} \tilde{F}(v)$ будет тождественным решению исходной задачи $v^* = Q^{-1}(\tilde{v}^*) = \arg\min\limits_{v \in X}(-F(v))$ в силу выражения (7).

Ранжирование полученных решений

После решения задач оптимизации из различных начальных точек формируется множество решений

$$\tilde{V}_{\text{opt}} = \{\tilde{v}_{\text{opt}.i} | \tilde{v}_{\text{opt}.i} = \arg \max_{v \in X^{n-1}} F(Q^{-1}(v)), \; i = 1, \ldots, \bar{i}\}.$$

В случае сложного характера функции, например, при ярко выраженной овражности, останов алгоритма оптимизации может происходить до достижения оптимальной точки. В связи с этим представляется целесообразным выделение оптимального по Парето множества $\tilde{X}_P^* \subseteq \tilde{X}^*$ перед ранжированием [5].

Ранжировать альтернативы $\tilde{v}_{\text{opt}.i}$, $i = 1, \ldots, \bar{i}$ по значению целевой функции $F(Q^{-1}(v))$ можно в режиме диалога с инвестором-ЛПР. В процессе диалога ЛПР по специальной процедуре предъявляются значения частных критериев качества, рассчитанные для каждого из решений.

Отметим, что диалоговые методы накладывают ряд требований, которым ЛПР должно удовлетворять:

— компетентность ЛПР в предметной области;

— объективность — на каждом шаге диалога ЛПР должно давать оценку качества каждого решения, объективно отражающую соответствие решения его целям и предпочтениям;

— точность, под которой понимается способность ЛПР сравнивать решения — оценки качества неэквивалентных решений должны различаться на величину, превышающую погрешность оценки ЛПР;

— последовательность (транзитивность) — ЛПР должно быть последовательно в своих решениях.

В соответствии со своими знаниями, предположениями и предпочтениями ЛПР анализирует предоставленную информацию и выбирает наилучшее решение. На этом процесс решения поставленной задачи (1)—(5) заканчивается.

ЗАКЛЮЧЕНИЕ

В статье изложена процедура решения задачи оптимизации финансового портфеля, предложен ряд приемов, позволяющих снизить размерность решаемой задачи оптимизации, ее вычислительную сложность, рассмотрены вопросы практического применения предлагаемого метода.

Не все компоненты портфеля инвестиций могут обладать неограниченной делимостью. Для обработки аргументов с дискретным множеством допустимых значений возможно решение задачи оптимизации для каждого из уровней каждой переменной с дискретным множеством значений. Фиксируя значение переменной, можно снизить размерность решаемой задачи оптимизации за счет увеличения числа ограничений типа равенств.

Создание эффективных методов оптимизации для решения задач смешанного типа (часть переменных принимает непрерывное, а часть — дискретное множество допустимых значений) — одно из наиболее важных и перспективных направлений дальнейшего развития методов решения задач оптимизации в предложенной постановке (1)—(5).

ЛИТЕРАТУРА

1. *Рыков А. С., Исходжанов Р. Р.* Многокритериальная модель оптимизации портфеля инвестиций // Тр. III междунар. конф. “Идентификация систем и задачи управления” SICPRO’2004. — М.: Ин-т пробл. управл., 2004.
2. *Рыков А. С.* Методы системного анализа: оптимизация. — М.: НПО «Издательство “Экономика”», 1999. — 255 с.
3. *Рыков А. С.* Поисковая оптимизация. Методы деформируемых конфигураций. — М.: Наука, 1993. — 216 с.
4. *Рыков А. С., Лановец В. В., Матвиенко М. Ю.* Система конструирования и исследования алгоритмов деформируемых конфигураций // Тр. междунар. конф. “Идентификация систем и задачи управления” SICPRO’2000. — М.: Ин-т пробл. управл., 2000.
5. *Ларичев О. И.* Теория и методы принятия решений. — М.: Логос, 2000. 296 с.

☎ *(095) 236-41-03*

E-mail: alexrykov@mail333.com

E-mail: rashidri@hotmail.com ❑

УДК 561.2.011.56

ИНФОРМАЦИОННОЕ УПРАВЛЕНИЕ В СОЦИАЛЬНО-ЭКОНОМИЧЕСКИХ СИСТЕМАХ: ЭЛЕМЕНТЫ УПРАВЛЕНИЯ И СПОСОБЫ ИНФОРМАЦИОННОГО ВОЗДЕЙСТВИЯ

Д. А. Кононов, В. В. Кульба, А. Н. Шубин

Институт проблем управления им. В. А. Трапезникова, г. Москва

Представлены результаты развития актуального направления исследований — информационного управления. Дан структурный анализ информационных элементов социально-экономической системы. Описаны механизмы и способы информационного управления.

ВВЕДЕНИЕ

Введенное в работе [1] понятие "информационной связи" между взаимодействующими объектами O, $P \in \mathbf{SO}$, где **SO** — совокупность рассматриваемых объектов, позволяет рассматривать потенциальную возможность их информационного взаимодействия. Механизм реализации этой возможности существенно зависит от свойств объектов O и P, а также от совокупности обстоятельств **SI** их отображения [2]. Так, для совокупности **SO**, в которую входят только объекты естественной природы, механизм реализации описывается ее объективными законами. Обстоятельства **SI** — те условия, при которых эти законы выполняются. Однако, как только в совокупности **SO** появляются объекты искусственной природы, механизм реализации информационного взаимодействия приобретает общественный характер и определяется, прежде всего, конкретной исторической деятельностью человека как субъекта действия.

Важная прикладная задача заключается в разработке моделей и методов обеспечения безопасного функционирования социально-экономической системы (СЭС) в условиях неблагоприятных информационных воздействий. Необходимо создать эффективную систему обеспечения информационной безопасности, которая в условиях неблагоприятных информационных воздействий могла бы обеспечить:

— качественный мониторинг развития СЭС (наблюдаемость);

— формирование спектра сценариев развития СЭС, описывающих все существенные тенденции и направления развития (полнота);

— формирование оптимального сценария развития СЭС (оптимальность);

— выполнение СЭС поставленных целей и задач (управляемость);

— перестройку и развитие структуры СЭС в соответствии с характеристиками внешней среды, а также управление информационной безопасностью на основе неполной априорной информации (адаптируемость).

Важнейшее требование к системе информационной безопасности состоит в оптимальном использовании на каждом этапе управления всех имеющихся ресурсов (информационных, экономических, политических, исторических, военных и др.) в текущей социально-экономической и политической ситуации. Построение оптимальных сценариев обеспечения информационной безопасности предполагает следующие этапы [3]:

— декомпозиция СЭС и процесса управления с точки зрения субъектно-объектного подхода;

— описание особенностей социальной системы как управляемой системы;

— выделение и описание социальных объектов и субъектов управления;

— выделение и описание структур социальных объектов и субъектов управления, в частности, стратификация и структуризация предметной области функционирования СЭС (экономической, правовой, социальной, технологической, военно-стратегической, политической...);

— определение факторов и показателей субъектов процесса функционирования СЭС;

— определение желательных состояний СЭС в процессе ее функционирования в терминах этих показателей;

— определение информационных совокупностей, описывающих процесс функционирования СЭС в каждой страте ее описания;

— определение структур информационных связей между ними;

— анализ информационного поля процесса функционирования и управления СЭС;

— определение и классификация информационных влияний на процесс функционирования СЭС;

— определение и классификация информационных угроз в процессе функционирования СЭС;

— определение совокупности альтернативных защитных реакций на информационные угрозы;

— сценарный анализ результатов информационных воздействий;

— сценарный синтез стратегий и способов информационной защиты.

Современные представления, которые могут быть положены в основу моделирования поведения социальных объектов управления, весьма разнообразны и существенно зависят от изучаемой СЭС. Совокупность учитываемых факторов, пригодных для эффективного описания, варьируется от биологических, климатических и психологических факторов (теория этногенеза Л. Гумилева [4]) до социально-экономических и политических (теория К. Маркса [5]). Процесс выделения существенных параметров и определения необходимых способов их измерения для социальных объектов управления представляет собой сложную задачу, решение которой может быть получено на основе объективных данных для конкретных социальных субъектов действия в соответствии с целями исследования.

УПРАВЛЕНИЕ В СОЦИАЛЬНО-ЭКОНОМИЧЕСКИХ СИСТЕМАХ

Последовательное изучение функционирования СЭС и процесса информационного управления методами прикладной математики приводит к необходимости уточнения исходных понятий и их формализованного описания.

Процесс принятия решения состоит из этапов выработки управленческого решения, в том числе анализа объекта управления. Он завершается, когда лицо, принимающее решение (ЛПР), четко формулирует *цели управления* и *способы их достижения*. Реализация принятого решения предполагает процесс воздействия на объект управления для достижения заданных целей и осуществляется с помощью посредника — *системы управления*, при этом априори процесс управления организуется для выполнения целей управления ЛПР. Совокупность совместно действующих объектов управления и системы управления называют *организационной системой*. На вербальном уровне под объектом управления подразумевается "пассивный" элемент организационной системы, а под системой управления — ее "активный" элемент. *Объектом управления* является пассивный элемент организационной системы, не имеющий собственных целей, отличных от целей управления. Поскольку ЛПР вырабатывает решение и реализует его через посредника — соответствующую часть системы управления, которая представляет собой *субъект действия*, то в ряде случаев (например, диверсия, коррупция и др.), возникает серьезное рассогласование между целями ЛПР и исполняющего эти решения субъекта. Это может приводить к искажению принятого к исполнению управленческого решения.

Субъектом действия является активный элемент системы, т. е. элемент, который может осознанно формулировать и достигать собственные цели.

Выделим социальные объекты СЭС, осуществляющие определенный вид деятельности и являющиеся носителем соответствующей формы общественного сознания.

Социальная группа как субъект действия может быть выделена по различным признакам общности интересов, например, классовым, национальным, профессиональным и т. п.; соответственно у нее есть свои *собственные интересы, выраженные в иерархии собственных целей*.

В отличие от субъектов действия типа "группа" выделяют субъект действия типа "масса" *Mas* — носителя *массового сознания*. Типичные примеры — участники широких политических или социокультурных движений (например, участники протеста против атомной угрозы), аудитории тех или иных средств и каналов массовой информации (например, читатели определенной газеты), потребители социально окрашенных (престижных, модных и т. п.) товаров и услуг, поклонники эстрадных звезд, "болельщики" футбольной команды. С точки зрения информационного управления важнейшее свойство массового сознания состоит в его возможности находиться в "клиповом состоянии", основная характеристика которого — поверхност-

ное усвоение и воспроизведение полученной информации.

Общественное мнение представляет собой состояние массового сознания, заключающее в себе отношение (скрытое или явное) людей к событиям и явлениям социальной действительности, к деятельности различных социальных институтов, групп и отдельных личностей [6].

Наряду с субъектами действия при моделировании управления в социальных системах рассматривают *социальные институты*, т. е. те элементы механизма управления, посредством которых принимается или проводится управляющее воздействие. Среди них выделим:

- *общественные организации* — организационные структуры, формально не подконтрольные государству как механизму управления;
- *институт общественного мнения* — "политический институт, который устойчиво и эффективно участвует в осуществлении власти, представляет собой один из признанных, узаконенных механизмов процесса принятия решений на всех уровнях жизни общества" [6];
- *средства массовой коммуникации* — организационные системы, которые обеспечивают возможность массового обмена информационными ресурсами общества, в том числе радио, почта, печать, телевидение, кино, эстрада, Интернет;
- *научная общественность* — независимый консультативный институт общества, который объединяет профессиональных экспертов по социально значимым проблемам.

Социальным субъектом действия назовем субъект действия или социальный институт СЭС. В процессе моделирования модели M_{MO} измерения его состояний и состояний его окружения M_{ME} следует рассматривать как преобразователи информации. На их вход поступает информация о внешних или внутренних параметрах социального субъекта действия, а на выходе формируется соответствующее мнение об истинных, по его оценке, значениях этих параметров. Таким образом, преобразователь M_{MO} формирует *внутреннее информационное поле субъекта действия*, а M_{ME} — его *внешнее информационное поле*. Эффективность преобразования зависит, в частности, от состояний M_{MO} и M_{ME}. Для социальных объектов такие изменения формируются в зависимости от субъективного представления самого субъекта действия.

"Пассивные" социальные объекты характеризуются набором расширенных состояний $\omega = (\chi, \nu) \in S(\Pi^{(S)})$, где χ — множество состояний внешней среды, ν — множество внутренних состояний системы S с концептом $\Pi^{(S)}$ [1], а "активные" требуют исходного описания в виде наборов тернарных отношений $\mathbf{o} = (\mathbf{I}, \mathbf{a}, \mathbf{O})$ между элементами, где $\mathbf{I} \subseteq 2^{S(\Pi^{(S)})}$ — обстоятельства ситуации, $\mathbf{a} \in Æ$ — предпринимаемое действие (стратегия). Результатом действия является новая ситуация $\mathbf{O} \subseteq 2^{S(\Pi^{(S)})}$. Субъект действия как системный элемент $E^{(A)}(B, r, p)$ с набором элементов B, связанных отношениями r с заданными свойствами p, активно участвует в функционировании системы информационного управления, что опосредованно может приводить к изменениям в его статусе [7]. Таким образом, СЭС представляется множеством систем $MS = \{O^{(i)}, i = \overline{1, n}; S^{(j)}, j = \overline{1, m}\}$ иерархически "переплетающихся" n объектов управления, а также m социальных субъектов действия, которые составляют K самостоятельных самоорганизующихся подсистем $SS^{(k)}$, $k = \overline{1, K}$. Термин "самоорганизующиеся" означает, что в процессе совместной деятельности ее участники могут вступать в новые, ранее не обусловленные в модели отношения, существенно меняющие как условия процесса управления, так и самой совместной деятельности.

Обобщая с позиций моделирования различные представления о поведении социальных объектов управления как элементов-организмов социально-экономической системы, можно говорить о "состояниях субъекта действия $S^{(j)}$, $j = \overline{1, m}$" или, более точно, о состоянии каждой из его окружающих подсистем $SS^{(k)}$, $k = \overline{1, K}$. При таком подходе совокупность всех возможных состояний СЭС представляет собой идентифицированную модель k-й подсистемы $M_O^{(k)}$, а совокупность всех мыслимых *состояний окружающей среды* — модель окружающей обстановки k-й подсистемы $M_E^{(k)}$. Обобщенная совокупность моделей $M_{OE}^{(k)} = (M_O^{(k)}, M_E^{(k)})$ описывает совокупность *расширенных состояний k-й подсистемы*, которые могут наблюдаться в расширенном фазовом пространстве [8].

Процесс управления социальными объектами осуществляется посредством механизма управления — определенной группы способов управления, применяемых для достижения заданных целей управления.

Механизмы административно-организационного управления — совокупность методов и средств достижения целей управления, опирающихся на волю и авторитет государства. Государство в качестве ЛПР создает специальные государственные институты для подготовки и реализации решений, создавая таким образом организационные системы.

Механизмы правового управления охватывают по существу всю систему и весь процесс управления.

С помощью права вносится необходимая, по мнению общества, мера нормативного регулирования, формальной определенности и упорядоченности, закрепляется объем полномочий и ответственности, т. е. устанавливаются всеобщие условия и правила для всех субъектов деятельности, взаимоотношений между ними, а также между другими элементами социальной системы. Для разработки и осуществления правового управления в обществе создаются специальные институты: законодательные органы, разрабатывающие и принимающие законодательные акты; судебные органы и прокуратура, компетентные выносить решение о мере наказания в соответствии с действующей системой законов, а также контрольные органы, надзирающие за соблюдением и точным их исполнением.

В отличие от административно-организационного управления, жестко предписывающего, регламентирующего определенные действия, *право* лишь устанавливает систему законодательных ограничений, в пределах которых возможны любые действия. Только в рамках *права* возможно говорить о "правах и обязанностях", "индивидуальной свободе", "взаимных обязательствах" и т. п., в том числе о правовом поле, в котором взаимодействуют социальные объекты.

Способ управления назовем *способом косвенного управления*, если управляющее воздействие, цель которого состоит в изменении состояния подсистемы $SS^{(k)}$, применяется к подсистеме $SS^{(r)}$ $(r \neq k)$. В указанном смысле к ним можно отнести:

- *механизмы социально-психологического управления* — совокупность средств и методов, основным объектом изучения и управления в которых является личность в коллективе; косвенность воздействия заключается в том, что, воздействуя на личность как субъект общения, взаимодействия, участника совместной деятельности, ЛПР через организационную систему реализует управление более сложным социальным объектом;
- *механизмы экономического управления* — совокупность мер, ориентированных на использование экономических интересов социальных субъектов действия; в основе этих методов лежит постулат о том, что каждый из этих субъектов действует так, чтобы максимально удовлетворить свои социально-экономические интересы, следовательно, с точки зрения ЛПР, следует создать такие правовые, организационные и прочие условия, которые стимулировали бы субъекта действия к достижению общей цели управления.

Центральным понятием информационного управления в рамках настоящей методологии является понятие информационной совокупности, объединяющее в единый объект искусственной природы наименование объекта Природы, его образ и обстоятельства I, в которых этот образ был получен [1]. Пусть задан механизм управления $\mathbf{Mu}(O^{(u)}, I)$, где $O^{(u)}$ — объект управления, I — обстоятельства применения механизма **Mu**.

Определение. *Описанием информационного поля механизма управления назовем совокупность*

$$\mathbf{Inf}(AdO^{(u)}, \mathbf{Mu}(O^{(u)}, I)), \qquad (1)$$

где $AdO^{(u)}$ — описание объекта управления.

Информационную совокупность (1) формируют для описания и последующего осуществления процесса управления, примененного к области описания $AdO^{(u)}$ объекта управления $O^{(u)}$. Специалисты по автоматизации управления такие данные представляют в виде *информационной модели управления*. При анализе взаимодействия ЛПР, объекта управления и системы управления на каждом из этапов управления выделим подробнее этапы формирования информационного поля управления.

На этапе **принятия решения** действует схема:

$$ЛПР \rightarrow СУ \rightarrow ОУ \rightarrow СУ \rightarrow ЛПР \rightarrow УР,$$

где $СУ$ и $ОУ$ — система и объект управления, $УР$ — управленческое решение. Схема описывает следующий порядок действий.

1. Определение ЛПР задания для системы управления по предоставлению информационной совокупности об объекте управления (информационное задание).

2. Выполнение системой управления информационного задания:

а) сбор данных об объекте управления;

б) создание информационной совокупности;

в) анализ информационной совокупности об объекте управления;

г) формулировка результатов анализа для "себя";

д) формулировка результатов анализа для ЛПР.

3. Передача запрошенной информационной совокупности ЛПР.

4. Формирование и принятие ЛПР управленческого решения.

Для каждого пункта рассмотренной схемы необходимо построить дерево угроз, описывающее состояние каждой из указанных процедур с точки зрения эффективности ее реализации для ЛПР. При этом можно выделить способы ослабления или устранения этих угроз. В частности, из рассмотренной схемы можно увидеть, что наиболее важным источником ошибок в принятии правильного решения для ЛПР является случай, когда система управления имеет собственные цели, отличные от цели управления. Тогда результаты п. 2, г могут сильно отличаться от результатов п. 2, д.

На этапе **реализации решения** действует схема:

$$УР \rightarrow СУ \rightarrow ОУ \rightarrow СУ \rightarrow ЦУ$$

(*ЦУ* — цель управления), которая описывает следующий порядок действий.

1. Формулировка информационного задания ЛПР для системы управления по реализации управленческого решения.

2. Выполнение системой управления информационного задания:

а) сбор данных об условиях реализации управленческого решения;

б) анализ информационной совокупности об условиях его реализации;

в) формулировка результатов анализа для "себя";

г) формулировка результатов анализа для ЛПР;

д) передача запрошенной информационной совокупности ЛПР.

3. Анализ реализации управленческого решения.

С учетом введенных понятий информационное управление — *совокупность способов воздействия на информационное поле механизма управления с заданными целями*. При этом информационное управление — способ воздействия, при котором информация есть отношение на совокупности данных [0, 3]. Таким образом, информация может являться объектом информационного воздействия и средством информационного воздействия, применяемого субъектом информационного воздействия.

В качестве *объекта информационного воздействия* при информационном управлении выступают *информационные совокупности объекта воздействия*, которые субъект информационного воздействия стремится изменить.

В качестве *средства воздействия* выступают *информационные совокупности субъекта информационного воздействия*, а также совокупность технологий (обстоятельства применения информационного управления), посредством которых проводится изменение информационной совокупности объекта воздействия.

Важным шагом в направлении модельного описания информационного воздействия на социальные объекты, стратегий и способов его реализации явилась принципиальная схема рефлексивного поведения человека, разработанная Лефевром [9—11]. Предполагается, что в психику человека как бы встроен врожденный информационный процессор. Его основная функция состоит в автоматической генерации внешних образов вместе с их субъективным миром. Работа этого процессора порождает специфический спектр человеческих реакций, никак сознательно не контролируемых и протекающих чрезвычайно быстро (одна-две миллисекунды). Этот вид рефлексии, в отличие от традиционной, был назван *быстрой рефлексией*.

Модель быстрой рефлексии Лефевра представляет субъекта, находящегося перед лицом биполярного выбора одной из двух альтернатив A и B. Одна из них, A, играет роль позитивного полюса, а другая, B — негативного. В простейшем случае, когда внутренний мир субъекта не содержит образов других субъектов, выбор может быть описан особой функцией $X_1 = f(x_1, x_2, x_3)$, где все переменные определены на интервале [0, 1]. Значение x_3 есть интенция субъекта совершить выбор позитивного полюса. Чем больше значение x_3, тем сильнее "желание субъекта совершить такой выбор". Значение X_1 интерпретируется как вероятность, с которой субъект готов выбрать позитивный полюс A в реальности. Значение x_1 есть давление внешнего мира в сторону позитивной альтернативы в момент выбора, а значение x_2 есть давление в сторону позитивной альтернативы, ожидаемое субъектом на основе его предшествующего опыта. Значения x_1 и $(1 - x_1)$ рассматривают как нормализованные полезности альтернатив A и B непосредственно в момент выбора, а значения x_2 и $(1 - x_2)$ как нормализованные ожидаемые полезности.

В ряде случаев математическое изучение функции $X_1 = f(x_1, x_2, x_3)$, зависящей от трех переменных, показывает, что она может быть представлена как композиция одной функции $F(x, y)$ от двух переменных: $X_1 = F(x_1, F(x_2, x_3))$, причем такое представление единственно. Тогда функцию $X_2 = F(x_2, x_3)$ интерпретируют как образ себя, имеющийся у субъекта. При такой интерпретации субъект обладает функционально правильным образом себя в прямом смысле этого слова, поскольку одна и та же функция, $F(x, y)$, описывает его одновременно как с собственной, так и с внешней точек зрения. Переменная x этой функции представляет перцептивный вход, а y — ментальный образ себя. Рассматривая функцию $F(x_2, x_3)$, которая играет роль образа себя, можно интерпретировать переменную x_2 как образ входа, а переменную x_3 как ментальный образ у образа себя. Этот вторичный образ называют *моделью себя*. При такой интерпретации переменная x_3 получает дополнительный смысл: она представляет не только интенцию, но и модель себя у субъекта. В результате имеем формальный аналог макроструктуры внутреннего мира социального объекта. Этот аналог есть структура композиции $F(x_1, F(x_2, x_3))$. В то же время эта структура описывает процесс когнитивных вычислений значения X_1: сначала вычисляется $X_2 = F(x_2, x_3)$, затем $X_1 = F(x_1, X_2)$.

Интенция субъекта может зависеть от факторов, которые не представлены переменными x_1 и x_2, и может принимать, в принципе, любое значение из интервала [0, 1]. В случае, когда интенция зависит только от x_1 и x_2, предполагается, что когнитивный механизм так координирует значения X_1 и x_3, чтобы выполнялось равенство $X_1 = x_3$. Это равенство соответствует утверждению, что субъект имеет *адекватную модель себя*.

Функция Лефевра, описывающая готовность субъекта выбрать позитивный полюс, такова:

$$X_1 = f(x_1, x_2, x_3) = x_1 + (1 - x_1)(1 - x_2)x_3, \quad (2)$$

где $x_1, x_2, x_3 \in [0, 1]$.

Тогда модель себя находится из функционального уравнения

$$\Phi(x_1, \Phi(x_2, x_3)) = x_1 + (1 - x_1)(1 - x_2)\, x_3, \quad (3)$$

где $x_1, x_2, x_3 \in [0, 1]$ и все значения $\Phi(x, y) \in [0, 1]$, имеющего единственное решение $\Phi(x, y) = 1 - y + xy = F(x, y)$. Следовательно, модель принятия решения субъектом представима как композиция

$$X_1 = F(x_1, F(x_2, x_3)), \quad (4)$$

а образу себя для функции (2) соответствует функция-модель

$$X_2 = F(x, y) = 1 - x_3 + x_2x_3.$$

Расширенная рефлексивная модель предполагает введение на трехмерном единичном кубе K^3 скалярных функций (2)—(4) непрерывной векторной переменной $\mathbf{x} = (x_1, x_2, x_3)$ [12].

В рамках этой модели можно поставить ряд задач информационного управления "клиповым" состоянием массового сознания субъекта *Mas*. Именно к этому состоянию применима модель быстрой рефлексии, так как главным свойством "клипового состояния" является свойство индивидуума "по мозговым извилинам мысль проскакивает по прямой". Выделим характерные особенности социального объекта типа "масса":

- статистический характер образующего его множества индивидов, состоящего из дискретных единиц, так что субъект *Mas* не представляет собой какого-либо самостоятельного, целостного образования, отличного от составляющих его элементов, т. е. не является системой;
- стохастическая (вероятностная) природа субъекта *Mas*, заключающаяся в том, что "вхождение" в него индивидов носит неупорядоченный, "случайный" характер, осуществляется по принципу "может быть, а может и не быть"; в результате такое множество всегда отличается размытыми, открытыми границами, неопределенным количественным и качественным составом;
- ситуативный характер субъекта *Mas*, связанный с тем, что оно образуется и существует исключительно на базе и в границах той или иной конкретной деятельности, невозможно вне ее, что делает субъект *Mas* относительно неустойчивым, временным образованием, меняющимся от случая к случаю;
- откровенно внегрупповая (или межгрупповая) природа субъекта *Mas*, проявляющаяся в том, что оно "разрушает" границы между всеми существующими в обществе социальными группами, отличается разнородным, "смешанным" социальным составом, включая в себя представителей различных классов и профессий, лиц с неодинаковым имущественным положением, образованием, уровнем культурного развития и т. д.;
- неспособность субъекта *Mas* выступать структурным элементом более широкого социального целого.

Резюме: субъект *Mas* является пассивным объектом управления, имеющим стохастический характер поведения, т. е. достаточно адекватно описываемый вероятностной моделью неопределенности.

"Как и сами массы, сознание в современных типах обществ возникает и формируется, в первую очередь, в процессе массовизации основных условий и форм жизнедеятельности людей (в сферах производства, потребления, общения, социально-политического участия, досуга), порождающем одинаковые или подобные устремления, интересы, потребности, навыки, склонности и т. д. С другой стороны, действие этих непосредственных условий и форм бытия закрепляется и получает свое дальнейшее завершение в производстве и распространении соответствующих видов массовой культуры, прежде всего, связанных с функционированием средств массовой информации и пропаганды. С их помощью указанные интересы, потребности, устремления широких слоев населения оформляются в виде серий одних и тех же образов действительности, способов познавательной деятельности и моделей поведения.

Будучи духовным продуктом объективных процессов, особого рода человеческой практики, массовое сознание оказывает активнейшее воздействие на многие стороны жизни общества, выступая в качестве важного регулятора форм поведения людей, в том числе посредством механизмов общественного мнения, общественного настроения" [6].

Пусть орган информационного управления имеет эталон поведения — число $\bar{X}_1 \in [0, 1]$, при этом основной управляющей переменной является x_1. Основная квазиинформационная гипотеза при сценарном анализе [3] информационного управления массовым сознанием на основе расширенной рефлексивной модели заключается в том, что переменная x_2 функционально зависит от векторной переменной $\mathbf{x}$. Вид соответствующей функциональной модели $x_2 = q(x_1, x_2, x_3)$ может выбираться на основе мониторинга СЭС или мнений экспертов.

"Информационное давление", т. е. применение управления x_1 без санкции объекта управления, может быть ограничено определенными условия-

ми: политическими, экономическими, техническими, культурными и т. п. Верхний и нижний "пороги давления" обозначим $\hat{x}_1$ и $\check{x}_1$.

Тогда формальная задача информационного управления "клиповым" состоянием массового сознания заключается в поиске решения задачи:

$$\rho(X_1 - \bar{X}_1) \to \inf, \tag{5}$$

$$0 \le \check{x}_1 \le x_1 \le \hat{x}_1 \le 1,\ 0 \le x_2 = q(x_1, x_2, x_3) \le 1,$$
$$X_1 = f(x_1, x_2, x_3) = x_1 + (1 - x_1)(1 - x_2)\, x_3,$$
$$\mathbf{x} \in K^3, \tag{6}$$

где ρ — метрика пространства R^3.

Конструирование различных метрических пространств, в том числе задание различных норм близости в целевой функции (5) и различного вида ограничений (6) позволяет исследовать различные варианты поведения массового сознания. В частности, естественно воспользоваться евклидовой нормой

$$\rho(X_1 - \bar{X}_1) = \|X_1 - \bar{X}_1\|_E. \tag{7}$$

Различные квазиинформационные гипотезы приводят к необходимости решения различных математических задач и различным режимам информационного управления.

Пусть $Mas^{(ad,o)}$ — множество объектов социального управления, находящихся в клиповом состоянии массового сознания и пользующихся для принятия решений адекватной моделью себя.

Теорема. *Задача оптимального управления социальным объектом $Mas^{(ad,o)}$ имеет решение.*

Доказательство. Пусть управляющий орган исходит из предположения адекватности управляемого субъекта. Это означает, что $X_1 = x_3$. Поскольку при условии $\check{x}_1 \ge 0$ из выражений (2)—(4) следует, что субъекту с адекватной моделью себя соответствует функция $X_1 = \dfrac{x_1}{x_1 + x_2 - x_1 x_2}$, то алгоритм решения задачи управления для метрики (7) заключается в поиске корня y уравнения

$$\frac{y(1-\bar{X}_1)}{\bar{X}_1(1-y)} = q\left(y, \frac{y(1-\bar{X}_1)}{\bar{X}_1(1-y)}, \bar{X}_1\right)$$
$$\text{и сравнения } \check{x}_1 \le y \le \hat{x}_1. \tag{8}$$

В противном случае решением будут "крайние управления" $\check{x}_1$ или $\hat{x}_1$.

Последовательность решения задачи управления заключается в мониторинге значений функции $q(x_1, x_2, x_3)$ с последующим поиском решения уравнения и сравнения (8).

СПОСОБЫ ИНФОРМАЦИОННОГО УПРАВЛЕНИЯ

Конкретные информационные воздействия на элементы СЭС можно классифицировать по следующим признакам: по источнику возникновения (внутри или вне СЭС); по длительности воздействия (разовые, периодические, постоянные); по природе возникновения (обусловленные текущей структурой СЭС, внешние воздействия на ее элементы).

Для определения и классификации информационных воздействий следует провести классификацию системных элементов СЭС и структуризацию объекта воздействия и субъекта влияния.

Для оценки эффективности конкретных способов информационного управления необходимо определить типовые задачи, решаемые с помощью информационного воздействия, провести анализ процесса формирования информационных операций и выработать критерии их оценки.

Среди наиболее серьезных по своим результатам задач, которые решаются с помощью информационного воздействия, выделим:

— создание атмосферы бездуховности и безнравственности, негативного отношения к культурному наследию СЭС противника;

— манипулирование общественным сознанием и политической ориентацией социальных групп населения страны с целью создания политической напряженности и хаоса;

— дестабилизация политических отношений между партиями, объединениями и движениями с целью провокации конфликтов, разжигания недоверия, подозрительности, обострения политической борьбы, провоцирование репрессий против оппозиции, провокация взаимоуничтожения;

— снижение уровня информационного обеспечения органов власти и управления, инспирирование ошибочных управленческих решений;

— дезинформация населения о работе государственных органов, подрыв их авторитета, дискредитация органов управления;

— провоцирование социальных, политических, национальных и религиозных столкновений;

— инициирование забастовок, массовых беспорядков и других акций политического протеста;

— затруднение принятия органами управления важных решений;

— подрыв международного авторитета государства, его сотрудничества с другими странами;

— нанесение ущерба жизненно важным интересам государства в политической, экономической, оборонной и других сферах;

— воздействие на вычислительные сети и автоматизированные системы управления.

Для решения этих задач необходимо осуществлять информационные воздействия на уровнях:

— выявления, создания и утилизации интерфейсных режимов информационных воздействий;

— разработки информационных операций и создания резерва ресурсов для их выполнения;

— осуществления информационного воздействия и поддержка интерфейсного режима;

— выхода из режима информационного воздействия.

Для повышения эффективности проведения операций в рамках информационного управления необходим тщательный анализ существующего информационного поля, что является основным условием для построения и осуществления целенаправленного информационного воздействия.

Интерфейсный режим информационного воздействия — совокупность каналов информационного воздействия и условий их активации. Под условиями активации канала информационного воздействия следует понимать совокупность ресурсов, необходимых для его поддержки, с одной стороны, и совокупность ресурсов для идентификации (в качестве пользователя) субъекта информационного воздействия, с другой. Наличие интерфейса информационного воздействия является необходимой предпосылкой осуществления информационного управления.

Особое значение информационных ресурсов обусловлено тем ключевым положением, которое они занимают в силу особой роли информации как системообразующего фактора. Они играют главенствующую роль по отношению к любым другим ресурсам государства — экономическим, научно-техническим и собственно военным. Очевидно, что эффективность информационных воздействий зависит от стоимости и номенклатуры используемых для организации информационного управления ресурсов. Вместе с тем воздействие на жизненно важнейшие информационные ресурсы способно при относительно малых затратах стабилизировать социально-экономическую структуру государства.

В качестве примеров основных интерфейсов информационного воздействия в масштабе государства можно указать следующие структуры: государственный строй (конституция, договорные отношения, властные структуры федерального и регионального уровней); правовая и судебная система; межгосударственные отношения; дипломатические структуры; общественные движения; партии; религия; морально-этические ценности; культурные институты и др.

Указанные структуры являются системообразующими для государства. Их постоянный мониторинг и анализ дают потенциальную возможность оказания враждебного информационного воздействия, которое может подаваться при определенных условиях на вход этих интерфейсных систем. Таким образом, государство в целом фактически является информационной самообучающейся системой. Поэтому для того, чтобы одна информационная самообучающаяся система была способна целенаправленно воздействовать на другую подобную систему, она должна ее понимать, т. е. знать ее структуру, а также ожидаемый результат при подаче определенного входного сообщения. Подбирая соответствующие входные данные, можно заставить информационную систему "смотреть на мир нужными глазами" и принимать неадекватные существующей социально-политической ситуации управленческие решения.

Известно, что любая информационная система обладает базовым набором знаний и связей между элементами, которые во многом определяют ее поведение. Чем больше мощность множества базовых элементов и их связей, тем устойчивее система к целенаправленному информационному воздействию. В то же время эффективность целенаправленного информационного воздействия возрастает, если оно сочетается с другими видами воздействия (эффект синергизма).

Устойчивость социально-экономических систем к информационному воздействию связана с их структурированностью и наличием множества дублирующих взаимосвязей. Поэтому для каждого выделенного интерфейса информационного воздействия необходимо определить порог его актуализации. Этот порог должен включать в себя необходимые ресурсы, а также сочетание благоприятных тематических, временных, территориальных и прочих сегментов информационных ресурсов государства.

Для государства можно ввести понятие уровня информационной управляемости. Резкое снижение этого уровня практически означает: сокращение информационных ресурсов СЭС, гибель элементов и подструктур, резкое снижение уровней экономической и социальной безопасности, ориентировка на решение ранее несвойственных политических и экономических задач.

Рассмотрим оценку эффективности ведения информационного управления с помощью средств массовой информации (СМИ). В современных условиях СМИ выступают в качестве основного информационного оружия "массового поражения", имеющего функции: информировать зрителя (читателя, радиослушателя); развлекать его; навязывать ему точку зрения своих хозяев или рекламодателей, т. е. в конечном итоге перепрограммировать его поведение в нужном направлении.

Основная характеристика СМИ — размер аудитории, которую данное средство охватывает своим информационным воздействием. Для печатного издания эта величина условно оценивается через

тираж, а для телевизионных и радиоканалов — через количество жителей на охватываемой ими территории вещания, обладающих необходимыми техническими средствами.

Однако для оценки эффективности СМИ как средства информационного управления одной такой характеристики недостаточно, так как люди, составляющие аудиторию СМИ, различаются по своим финансовым, умственным и социальным возможностям. Поэтому СМИ обладают различными возможностями как по доведению информации, так и по эффективности информационного воздействия в разных слоях населения (политическая, военная и культурная элита, финансовые круги, средний класс, население, живущее за чертой бедности и т. д.). Оценивая СМИ, необходимо учитывать не только его возможности по охвату населения, но и потенциальные возможности воздействия на разные слои населения.

Первые две выделенные функции СМИ в основном нацелены на то, чтобы делать их пользующимся спросом товаром, который приносит прибыль владельцам в соответствии с законом прибавочной стоимости. Очевидно, что лишь третья функция СМИ, связанная с эффективным информационным воздействием на население, делает их серьезным средством информационного управления.

ЗАКЛЮЧЕНИЕ

Представленный анализ информационных элементов [1], способов и механизмов информационного управления позволяет приступить к моделированию систем информационной безопасности различных структурных и внеструктурных элементов изучаемой СЭС. При этом следует обратить внимание на решение следующих проблем:

— определение качественно различных состояний социальных объектов управления как объектов информационного воздействия;

— определение силы информационного воздействия как “физической величины”, характеризующей изменение состояния объекта целенаправленного информационного воздействия в сравнении с его исходным состоянием;

— выявление характеристик и свойств введенной “физической величины” по отношению к различным социальным объектам управления;

— разработка практических способов измерения эффективности информационного воздействия.

В основе решения этих вопросов лежит понятие *информационной акции*, которое будет рассмотрено в следующей статье.

5ЛИТЕРАТУРА

1. *Кононов Д. А., Кульба В. В., Шубин А. Н.* Информационное управление: формализованное описание информа ционных элементов // Проблемы управления. — 2004. — № 2. — С. 45—51. (www.ipu.ru/period/pu)
2. *Кульба В. В, Малюгин В. Д., Шубин А. Н.* Информационное управление (предпосылки, методики и средства) // Проблемы управления. — 2003. — № 1. — С. 62—67. (www.ipu.ru/period/pu)
3. *Кульба В. В., Кононов Д. А., Косяченко С. А., Шубин А. Н.* Методы формирования сценариев развития социально-экономических систем. — М.: СИНТЕГ, 2004. — 296 с.
4. *Гумилев Л.* Этносфера: История людей и история природы. — М.: Экопрос, 1993. — 544 с.
5. *Платонов С.* После коммунизма: Кн., не предназнач. для печати — Второе пришествие: беседы. — М.: Молодая гвардия, 1991.
6. *Грушин Б.* 50/50: Опыт словаря нового мышления. — М.: Прогресс, 1989. — С. 204, 214.
7. *Логика* и методология системных исследований. — Киев — Одесса, Вища школа. — 1977.
8. *Кононов Д. А., Кульба В. В., Ковалевский С. С., Косяченко С. А.* Синтез формализованных сценариев и структурная устойчивость сложных систем (синергетика и аттрактивное поведение). Препринт. — М.: ИПУ РАН, 1998.
9. *Lefebvre V. A.* An Algebraic Model of Ethical Cognition // Journal of Mathematical Psychology. — 1980.
10. *Lefebvre V. A.* (1987) The Fundamental Structures of Human Reflexion. // Journal of Social and Biological Structures. — 1987. — 10 129 175.
11. *Рефлексивные* процессы и управление // Международный научно-практический журнал. — 2001. — Т. 2, № 1.
12. *Kononov D. A., Kul'ba V. V, and Shubin A. N.* Scenario analysis of information conflicts // 10th IFAC Conf. on Technology and International Stability; SWIIS'03, 3—4 July, 2003; Waterford, Republic Ireland, Preprints Volume. — 2003. — P. 36—40.

☎ *(095) 334-90-09* ❑

ВНИМАНИЮ ЧИТАТЕЛЕЙ!

Журнал **"ПРОБЛЕМЫ УПРАВЛЕНИЯ"** входит в Перечень периодических научных изданий, рекомендуемых ВАК для публикации научных работ, отражающих основное научное содержание докторских диссертаций.

УДК.658.314.7:330.115

МЕХАНИЗМЫ МАНИПУЛИРОВАНИЯ КОРПОРАЦИЕЙ

И. А. Агеев*, И. В. Гурлев**, В. В. Цыганов**

*Компания "РОЭЛ Консалтинг", г. Москва
**Институт проблем управления им. В. А. Трапезникова, г. Москва

Найдены условия прогрессивности и оптимальности механизмов управления эволюцией корпорации в стохастической среде, предусматривающих самообучение классификации. Дан анализ механизмов манипулирования, применяемых в процессе технократического управления эволюцией организации. Рассмотрено манипулирование первого рода, воздействующее на эмоции заинтересованного лица-Ученика, и второго рода, использующее количественные показатели и рациональность самообучающегося заинтересованного лица-Ученого. Показано, что найдется такое манипулирование, что корпоративный механизм будет регрессивным.

ВВЕДЕНИЕ

В работе [1] рассмотрены задачи управления эволюцией организации в стохастической среде, предполагающего применение процедур обучения опознаванию образов с использованием указаний учителя. Часто роль учителей в организациях выполняют профессиональные эксперты. Осуществляя технические функции, они приобретают власть, и поэтому их называют технократами, а управление, основанное на их рекомендациях — технократическим. В работе [1] определены условия, при которых технократический механизм приводит к развитию организации. Показано, что технократический механизм с лжеучителем, манипулирующим указаниями, регрессивен. В настоящей работе продолжен анализ механизмов манипулирования, применяемых в процессе технократического управления эволюцией организации.

МЕХАНИЗМЫ САМООБУЧЕНИЯ

Массовое применение информационных технологий изменения сознания привело к возникновению адаптивных и самообучающихся организаций [2]. Эффективность воздействия информационных технологий на их членов — заинтересованных лиц — весьма различна: от нулевой (индифферентность) до максимальной (полное подчинение указаниям учителя). В работе [1] рассмотрена двухуровневая система "центр—дальновидный элемент (ДЭ)". Роль центра играет обучаемое экспертом-учителем заинтересованное лицо (ученик), а роль ДЭ — работник, исполнитель, человек дела.

Предположим теперь, что в роли центра выступает заинтересованное лицо, которое самостоятельно обучается классификации наблюдаемых ситуаций (для краткости будем называть такой центр — Ученый). В стохастической обстановке Ученый настраивает свое решающее правило так, чтобы минимизировать потери при классификации. Настройка осуществляется по наблюдениям входа и выхода организации с помощью процедуры самообучения. Решающее правило определяет стимулирование дальновидного элемента. Поскольку в адаптивном механизме результаты настройки используются для принятия решений, ДЭ может влиять на стимулы, работая с разной эффективностью.

Рассмотрим двухуровневую систему "Ученый—ДЭ", в которой центр формирует оценку a_t параметра решающего правила на основе процедуры самообучения I и стимулирует ДЭ на основе её сопоставления с выходом ДЭ y_t. Как и в работе [1], ξ — случайная величина, характеризующая потенциал ДЭ, $\xi \in \Delta \subset R^1$. Задача Ученого заключается в классификации ситуации ξ путем отнесения ее к одной из двух областей, составляющих множество Δ. Неправильная классификация приводит к потерям. Предположим вначале, что Ученому известна плотность $q(\xi)$ распределения величины ξ. Обозначим через Δ_1 и Δ_2 некоторое разбиение множества Δ на две области, $\bigcup_{k=1}^{2} \Delta_k = \Delta$. При классификации, т. е. отнесении ситуации ξ к одной из этих

областей, Ученый принимает решение, связанное с некоторым риском. Проблема заключается в определении разбиения, минимизирующего средний риск, связанный с классификацией. Введем для каждой, пока неизвестной области Δ_k, $k = 1, 2$, функции потерь $F_k(c, \xi)$, где c — неизвестный параметр. Минимизируется средний риск, оценивающий качество классификации

$$J(c) = \sum_{k=1}^{2} \int_{\Delta_k} F_k(c, \xi) q(\xi) d\xi \underset{c}{\rightarrow} \min \qquad (1)$$

Условие минимума среднего риска (1) имеет вид:

$$M_\xi \left\{ \sum_{k=1}^{2} B_k(c, \xi) \frac{dF(c, \xi)}{dc} \right\} = 0,$$

$$B_k(c, \xi) = \begin{cases} 1 \text{ при } \xi \in \Delta_k, \\ 0 \text{ при } \xi \notin \Delta_k. \end{cases} \qquad (2)$$

Принадлежность ситуации ξ области Δ_k определяется знаком решающего правила

$$\mu_{12}(c, \xi) = F_1(c, \xi) - F_2(c, \xi): \xi \in \Delta_1,$$
$$\text{если } \mu_{12}(c, \xi) < 0 \text{ и } \xi \in \Delta_2 \text{ если } \mu_{12}(c, \xi) \geq 0. \qquad (3)$$

Полагая $F_1(c, \xi) = \xi - vc$, $v < 1$, $F_2(c, \xi) = d(c - \xi)$, $vc \leq \xi \leq c$, из выражения (3) получаем решающее правило при классификации в виде

$$\xi \in \Delta_1, \text{ если } \xi < \frac{d+v}{d+1} c, \text{ и } \xi \in \Delta_2, \text{ если } \xi \geq \frac{d+v}{d+1} c,$$

где параметр c решающего правила может быть найден как решение задачи (1), (2), причем $q(\xi)$ известно Ученому, $d > 0$ — коэффициент.

Предположим теперь, что плотность распределения $q(\xi)$ неизвестна Ученому. В этом случае непосредственное определение параметра c как решения задачи оптимизации (1), (2) невозможно. Возникает необходимость в настройке параметра с решающего правила по наблюдениям ситуации ξ_t для минимизации среднего риска (1). Процедура настройки параметра решающего правила имеет вид

$$c_{t+1} = I^k(c_t, \xi_t) = \begin{cases} c_t + \gamma v \text{ при } \xi_t < \dfrac{d+v}{d+1} c_t \\ c_t - \gamma d \text{ при } \xi_t \geq \dfrac{d+v}{d+1} c_t. \end{cases} \qquad (4)$$

Предполагается, что оценки, получаемые с помощью процедуры (4), сходятся к оптимальному значению c^*:

$$c_t \underset{c}{\rightarrow} c^* = \arg\min_c J(c). \qquad (5)$$

Рассмотрим теперь задачу оптимального синтеза самообучающегося механизма функционирования $\Sigma^k = (I^k, f)$, в котором оценки параметра решающего правила определяются с помощью процедуры самообучения (4):

$$a_{t+1} = I^k(a_t, y_t), \qquad (6)$$

где y_t — наблюдаемое Ученым состояние ДЭ, не обязательно совпадающее с его потенциалом ($y_t \leq \xi_t$), f — процедура стимулирования. Стимул $\varphi_t = f(a_t, y_t)$ в периоде t формируется на основе текущей оценки параметра решающего правила (6). Как и в работе [1], целевая функция ДЭ имеет вид:

$$V_t = \sum_{\tau=t}^{t+T} \rho^\tau \varphi_\tau, \qquad (7)$$

где ρ — коэффициент дисконтирования, используемый для приведения будущих стимулов к текущему моменту времени t, $0 < \rho < 1$; T — дальновидность ДЭ, исчисляемая в периодах времени. Прогноз этой целевой функции $\hat{V}_t(a_t, y_t)$ в периоде t, в зависимости от a_t, y_t, определяется согласно формуле (6) работы [1]. Предполагается также благожелательность ДЭ по отношению к центру [1].

Будем говорить, что механизм $\Sigma^k = (I^k, f)$ прогрессивен, если $y_t^* = \xi_t$, и оптимален, если оценки параметра, получаемые на основе стохастической процедуры (6), сходятся к c^* : $a_t \underset{t}{\rightarrow} c^*$.

Теорема 1. *Для прогрессивности и оптимальности самообучающегося механизма функционирования $\Sigma^k = (I^k, f)$ с процедурой обучения классификации (6) достаточно, чтобы*

$$f(a_t, y_t) = \Theta(y_t - a_t(d + v)/(d + 1)) =$$
$$= \begin{cases} 1 \text{ при } y_t \geq a_t(d+v)/(d+v) \\ 0 \text{ при } y_t < a_t(d+v)/(d+1). \end{cases} \qquad (8)$$

Доказательство. По условию (8) с ростом показателя y_t текущий стимул ДЭ $\varphi_t = f(a_t, y_t)$ возрастает. Ученый использует процедуру самообучения (6), при которой оценки a_τ убывают с ростом показателя y_t при $\tau = \overline{t+1, t+T}$. Следовательно, согласно условию (8) будущие стимулы ДЭ $\varphi_\tau = f(a_\tau, y_\tau)$ возрастают с ростом показателя y_t при $\tau = \overline{t+1, t+T}$. Далее, целевая функция V_t монотонно возрастает по $\varphi_\tau = f(a_\tau, y_\tau)$, $\tau = \overline{t, t+T}$. Но стимул φ_t монотонно возрастает по y_t. Следовательно, с ростом показателя y_t возрастает и прогнозное значение целевой функции $\hat{V}_t(a_t, y_t)$. По-

скольку $y_t \le \xi_t$, то максимум $\hat{V}_t(a_t, y_t)$ достигается при $y_t = \xi_t$. Следовательно, согласно формуле (8) работы [1], $\xi_t \in R(\Sigma^k, \xi_t) = \operatorname{Arg}\max_{y_t \le \xi_t} \hat{V}_t(a_t, y_t)$. В силу благожелательности ДЭ по отношению к центру, $y_t^* = \xi_t$. Таким образом, механизм $\Sigma^k = (I^k, f)$ прогрессивен и выполняется условие (5), что и требовалось доказать.

Теорема 1 допускает простую интерпретацию. Ученый наблюдает выход y_t ДЭ в периоде t, $y_t < \xi_t$, где ξ_t — неизвестный случайный потенциал. На основе этого он формирует оценки параметра решающего правила (6). Каждый шаг настройки этого параметра означает изменение восприятия, осознания Ученым наблюдаемой ситуации. Далее, в соответствии с принятым решающим правилом, Ученый классифицирует ДЭ по результатам (показателям) ДЭ. Именно, при $y_t < x_t$ ДЭ относится к классу неблагополучных, в противном случае — к благополучным. Любое из этих решений связано для Ученого с определенным риском. В первом случае потери F_1 возрастают при увеличении эффективности ДЭ (например, несправедливое наказание работника). Во втором случае эти потери F_2 возрастают с ухудшением показателя y_t (незаслуженное премирование работника). Величина $x_t = (d + v)a_t/(d + 1)$ соответствует нижней границе удовлетворительной работы. При этом оценки a_t сходятся к оптимальному значению c^*.

ДЕМОКРАТИЯ И ПРОГРЕСС КОРПОРАЦИИ

Неотъемлемой частью организации является сообщество ее членов — заинтересованных лиц. Оно выступает субъектом управления в процессе эволюции социально-экономических систем [2]. Пример — корпорация, субъектом управления которой является сообщество акционеров, а объектом — наемный менеджер, осуществляющий оперативное руководство текущей деятельностью. Рассмотрим механизмы адаптивного управления эволюцией корпорации, в котором есть самообучающиеся и обучаемые заинтересованные лица. Их обучение основано на процессе формирования определенной системы норм, позволяющих им функционировать в качестве членов сообщества. Обучение требует специальных знаний и экспертов.

Механизм функционирования организации, в которой в качестве центра выступают сообщество Ученого и Ученика с Учителем, а в качестве ДЭ — менеджер, назовем его корпоративным. Исследование этого механизма включает в себя анализ процедур индивидуального выбора Ученого и Ученика. Затем коллективный выбор сообщества рассматривается как совокупность индивидуальных выборов. Обратная связь со стороны сообщества формирует стимулы ДЭ. Менеджер поощряется заинтересованными лицами в зависимости от классификации результатов его деятельности, которая, в свою очередь, зависит от соотношения выхода корпорации y_t (например, дивидендов) и нормы ("планки ожиданий").

Рациональное заинтересованное лицо минимизирует риск, связанный с ожиданиями, исходя из текущих результатов функционирования менеджера y_t. Для этого i-е заинтересованное лицо применяет ту или иную процедуру обучения и формирует норму x_{it}, $i = 1, 2$. После этого оно использует процедуру решения $f_i(x_{it}, y_t)$. Если заинтересованное лицо принимает решение о поощрении менеджера, то целевая функция (7) последнего увеличивается на 1: $\varphi_{it} = f_i(x_{it}, y_t) = 1$, в противном случае она не увеличивается: $\varphi_{it} = f_i(x_{it}, y_t) = 0$. Нормативная процедура принятия решения имеет вид:

$$f_i(x_{it}, y_t) = \Theta(y_t - x_{it}) = \begin{cases} 1, \text{ если } y_t \ge x_{it} \\ 0, \text{ если } y_t < x_{it}. \end{cases} \qquad (9)$$

Корпоративный механизм $\Sigma^P = (I_1, f_1, I_2, f_2)$ представляет собой совокупность процедур оценивания I_1 и решения f_1, используемых Ученым, а также процедур оценивания I_2 и решения f_2, используемых Учеником. Результаты решения в периоде t объединяют в стимул менеджера $\varphi_t = \Phi(\varphi_{1t}, \varphi_{2t})$, где Φ — монотонно возрастающая функция агрегирования. Стимулы определяют целевую функцию менеджера (7). Дальновидный менеджер выбирает состояние y_t в периоде t так, чтобы её максимизировать.

Теорема 2. *Если заинтересованные лица используют процедуру самообучения классификации (4) или процедуру обучения опознавания образов I^S (8) работы [1], то корпоративный механизм $\Sigma^P = (I^k, \Theta, I^S, \Theta)$ прогрессивен.*

Доказательство. Из теоремы 1 следует, что если заинтересованное лицо-Ученый использует процедуру самообучения классификации (4) и процедуру решения Θ (9), то соответствующий механизм $\Sigma^k = (I^k, \Theta)$ прогрессивен. Далее, из теоремы 1 работы [1] следует, что если заинтересованное лицо-Ученик использует процедуру обучения опознаванию образов I^S, определяемую согласно формуле (8) работы [1], и процедуру решения (9), то соответствующий механизм $\Sigma^S = (I^S, \Theta)$ прогрессивен. Но тогда, в силу монотонности функции агрегирования Φ, корпоративный механизм $\Sigma^P = (I^k, \Theta, I^S, \Theta)$, как композиция прогрессивных механизмов Σ^k и Σ^S, прогрессивен.

Теорема 2 определяет механизм, при котором соблюдение индивидуумами личных интересов приводит к росту благосостояния их сообщества. Она характеризует условия, при которых демократия обеспечивает прогресс корпорации. Самообучающееся заинтересованное лицо, решая задачу минимизации риска, использует процедуру обучения классификации (4). При этом оно формирует норму b_t в периоде t. Если оказывается, что $y_t \geqslant b_t$, то оно поощряет менеджера ($\varphi_{1t} = 1$), в противном случае — нет ($\varphi_{1t} = 0$). Заинтересованное лицо-Ученик, решая задачу минимизации риска, использует процедуру обучения опознаванию образов, основанную на указаниях эксперта. В процессе обучения уточняется норма a_t. Если оказывается, что $y_t \geq a_t$, то поощрение $\varphi_{2t} = 1$, в противном случае $\varphi_{2t} = 0$.

Если заинтересованные лица наблюдают и учатся самостоятельно или с помощью экспертов, а последние объективны (правдивы) в том смысле, что сообщают истинную оценку положения дел и результатов функционирования корпорации, то демократия обеспечивает прогресс сообщества в целом, например, рост уровня потребления заинтересованных лиц. Таким образом, демократия обеспечивает прогресс корпорации, если ДЭ действительно дальновиден, заинтересованные лица рациональны (минимизируют риск, связанный с классификацией), а Учитель правдив. Поэтому теорему 2 можно назвать теоремой о демократии и прогрессе.

МАНИПУЛИРОВАНИЕ ПЕРВОГО И ВТОРОГО РОДА

Рассмотрим возможности манипулирования и сдвига экспертом-технократом оценок параметров решающего правила, т. е. изменения норм индивидуального сознания Ученого и Ученика. Важное место здесь занимает процедура (10) (см. работу [1]), которую можно назвать процедурой манипулирования сознанием Ученика. С ее помощью лжеучитель влияет на дальновидного менеджера (ДЭ), меняя знак корпоративной обратной связи с положительной на отрицательную, и наоборот. В технократическом механизме для этого используются указания $L_1 = 1$ и $L_2 = 0$ (см. формулу (10) в работе [1]), которые можно трактовать как противоположности — "хорошо" (1) и "плохо" (0), "добро" (1) и "зло" (0) и т. д. Процесс эволюции корпорации подчиняется диалектическому закону единства и борьбы противоположностей. Поэтому в каждом событии имеются два противоположных аспекта. Меняя акценты, технократ усиливает один и ослабляет другой. С этим связано манипулирование первого рода. В нем воздействие направлено на эмоции заинтересованного лица-Ученика. Оно облекается в форму простой дихотомии — "хорошо" ($L = 1$) или "плохо" ($L = 0$).

Манипулирование второго рода использует рациональность, логику самообучающегося заинтересованного лица — Ученого. В отличие от манипулирования первого рода, здесь используются количественные показатели, характеризующие противоположные аспекты эволюции корпорации (например, затраты и результаты). Проще и нагляднее использовать в сообщении один показатель, допускающий прямое сравнение положения дел. Такой универсальный показатель, например, в экономике — деньги, в политике — рейтинги. Формально манипулирование второго рода — это манипулирование данными, поступающими на вход Ученого, в реальном масштабе времени (т. е. синхронно с выходным показателем ДЭ). Первый способ — введение поправки или постановка помехи, корректирующей сигнал с выхода ДЭ. Например, наблюдаемый заинтересованным лицом выход ДЭ y_t^* в периоде t

$$y_t^* = y_t - D_t, \qquad (10)$$

где y_t — истинный выход ДЭ в периоде t, D_t — поправка, устанавливаемая технократом для искажения входных данных Ученого. Другой способ манипулирования Ученым — усиление или ослабление истинного выхода. На практике для манипулирования второго рода могут использоваться специальные механизмы. Например, получаемые в результате выборочных опросов заинтересованных лиц частные рейтинги нередко применяются для искажения реальных результатов деятельности дальновидного менеджера. Аналогичным образом могут использоваться спекулятивные колебания цен акций корпорации на фондовом рынке и т. п.

ДУАЛЬНОЕ МАНИПУЛИРОВАНИЕ

Дуальное манипулирование – это одновременное манипулирование первого и второго рода. В корпоративном механизме с дуальным манипулированием в роли эксперта выступает технократ, который сообщает заинтересованным лицам (Ученику и Ученому) указания и данные, исходя из собственных интересов, а не на основе истинного положения дел. Например, технократ может присваивать разницу $\xi_t - y_t$, т. е. неиспользуемые "резервы" корпорации. В этом случае он заинтересован не в увеличении, а в уменьшении значения y_t до минимально допустимого уровня $p_t (y_t \geq p_t)$. В механизме дуального манипулирования корпорацией технократ, во-первых, является лжеучителем и дает Ученику ложные указания $L(y_t)$, определяемые по формуле (10) в работе [1] ($L_t = 0$, если работа в периоде t удовлетворительная, и $L_t = 1$, ес-

ли работа неудовлетворительная). Во-вторых, технократ корректирует данные, поступающие к Ученому, на величину $D_t = D(y_t)$, $t = 1, 2, ...$ Механизм, обеспечивающий выполнение условий $y_t = p_t$, $t = 1, 2, ...$, называется регрессивным.

Теорема 3. *Найдется дуальное манипулирование в виде последовательности* $\{L_t, D_t, t = 1, 2, ...\}$ *такой, что корпоративный механизм* $\Sigma^P = (I^k, \Theta, I^L, \Theta)$, *в котором используются наблюдаемые состояния ДЭ (10), будет регрессивным.*

Доказательство. Согласно теореме 1, если заинтересованное лицо использует процедуру самообучения классификации (4) и процедуру решения (9), то механизм $\Sigma^k = (I^k, \Theta)$ прогрессивен, т. е. обеспечивает рост целевой функции ДЭ (7) с ростом наблюдаемого Ученым выхода y_t^8. Положим в процедуре (10) $D_t = (1 + \alpha)y_t - d$, причем $\alpha > 0$, $d \geq \alpha\xi_t$, $t = 1, 2, ...$ Тогда, согласно выражению (10), $y_t^* = d - \alpha y_t$. При этом из теоремы 1 следует, что механизм $\Sigma^k = (I^k, \Theta)$ обеспечивает уменьшение целевой функции ДЭ (7) с ростом истинного выхода y_t. Поэтому механизм $\Sigma^k = (I^k, \Theta)$ обеспечивает выполнение условий $y_t = p_t$, $t = 1, 2, ...$, т. е. является регрессивным.

Далее, в соответствии с теоремой 2 работы [1], технократический механизм с лжеучителем $\Sigma^L = (I^L, \Theta)$, в котором используется процедура стимулирования (9) и процедура оценки (11) в указанной работе (причем $\xi^* > b_t$, $t = 1, 2, ...$) регрессивен. Но тогда, в силу монотонности функции агрегирования Φ, корпоративный механизм $\Sigma^P = (I^k, \Theta, I^L, \Theta)$, как композиция регрессивных механизмов Σ^k и Σ^L, регрессивен, что и требовалось доказать.

Наряду с процедурой (10), простой способ манипулирования Ученым — усиление или ослабление истинного выхода: $y_t^* = k_t y_t$, где k_t — коэффициент усиления (при $k_t > 1$) или ослабления (при $k_t < 1$) истинного выхода. Нетрудно показать, что найдутся последовательности коэффициентов $\{k_t\}$, при которых соответствующий механизм манипулирования регрессивен. Применяя этот подход, можно, например, манипулировать данными о дивидендах y_t заинтересованных лиц путем введения коэффициентов ослабления (или усиления) k_t, аргументируя это необходимостью приведения к реальным ценам, с учетом инфляции. Тогда k_t — это поправка на инфляцию. Поскольку рост цен на разные товары неодинаков, всегда имеется неопределенность в отношении инфляции: $k_t \in [l_t, l_t + \Delta_t]$ и можно усилить или ослабить влияние эффекта роста дивидендов на заинтересованных лиц.

Таким образом, технократ может эффективно воздействовать на дальновидного менеджера путем дуального манипулирования сознанием заинтересованных лиц, меняя знак корпоративной обратной связи с положительной на отрицательную.

Предположим, что результаты деятельности менеджера характеризуются несколькими скалярными показателями y_{jt}, $j = \overline{1, J}$, а его стимулирование осуществляется на основе агрегирования локальных оценок деятельности по каждому показателю: $f_{jt} = f_j(y_{jt})$, $\varphi_{jt} = \varphi_j(y_{jt})$, $j = \overline{1, J}$. Комплексные оценки f_t^*, φ_t^*, получаемые в результате агрегирования, являются монотонно возрастающими функциями локальных оценок деятельности: $f_t^* = F(f_{1t}, ..., f_{Jt})$, $\varphi_t^* = G(\varphi_{1t}, ..., \varphi_{Jt})$. Тогда комплексный корпоративный механизм можно рассматривать как композицию независимых локальных корпоративных механизмов, каждый из которых основан на одном скалярном показателе y_{jt}, $j = \overline{1, J}$.

Согласно теореме 2, технократ может мотивировать менеджера к раскрытию потенциала ξ_{mt} по тому или иному показателю y_{mt}, $1 < m < J$, с помощью сообщения заинтересованным лицам правдивых указаний. И наоборот, давая им ложные указания и данные по показателю y_{kt}, $1 < k < J$, технократ может мотивировать менеджера к сокрытию потенциала ξ_{kt} по показателю y_{kt}. Действительно, согласно теореме 3, соответствующий локальный корпоративный механизм, основанный на показателе y_{kt}, регрессивен. Таким образом, технократ может обеспечивать прогресс в одном аспекте деятельности корпорации, характеризуемом показателем y_{kt}, и регресс — в другом, характеризуемом показателем y_{mt}. Тем самым, корпорация шаг за шагом "разворачивается" в пространстве показателей, причем эволюция происходит в направлении, определенном технократом. Таким образом, путем манипулирования общественным сознанием осуществляется технократическое управление эволюцией корпорации.

ЛИТЕРАТУРА

1. *Агеев И. А., Ермошкин А. И., Цыганов В. В.* Механизмы технократического управления эволюцией организации // Проблемы управления. — 2004. — № 2. — С. 40—44.
2. *Цыганов В. В., Бородин В. А., Шишкин Г. Б.* Интеллектуальное предприятие: механизмы овладения капиталом и властью. — М.: Университетская книга, 2004. — 768 с.

☎ *(095) 194-94-85*

E-mail: igor.ageev@g9.relcom.ru

☎ *(095) 334-79-00*

E-mail: bbc@ipu.ru ❑

УДК 001.8+007.5

СТРУКТУРНО-ПАРАМЕТРИЧЕСКОЕ МОДЕЛИРОВАНИЕ И ИДЕНТИФИКАЦИЯ АНОМАЛЬНЫХ СИТУАЦИЙ В СЛОЖНЫХ ТЕХНОЛОГИЧЕСКИХ СИСТЕМАХ

Ю. А. Ивашкин

Московский государственный университет прикладной биотехнологии

Рассмотрены матричные модели структурно-сложных ситуаций взаимодействия в больших системах и алгоритмы их идентификации и прогнозирования.

Предлагаемый подход к моделированию и идентификации аномальных ситуаций в больших системах основан на формализованном описании в матричной форме влияния различных факторов на критерий достижения цели, выражаемый многомерной суммой взвешенных нормированных отклонений параметров состояния системы от заданных значений [1, 2].

Эффективность функционирования системы любой физической или социальной природы описывается вектором параметров $Y = \{y_1, ..., y_m\}$ или критерием $Q(y_1, ..., y_m)$ оценки состояния системы (качество продукции, производительность, себестоимость, КПД и т. п.). Компоненты вектора Y в общем случае являются функциями характеристик входных потоков $G = \{g_1, ..., g_r\}$ и параметров состояния системы $X = \{x_1, ..., x_n\}$, которые, в свою очередь, зависят от факторов возмущения $V = \{v_1, ..., v_q\}$ и управляющих воздействий $U = \{u_1, ..., u_p\}$.

Изменение состояния системы в первом приближении можно описать линейными уравнениями в приращениях

$$Q(Y) = \sum_{i=1}^{m} q_i \Delta y_i$$

$$\Delta y_i = \sum_{j=1}^{n_x} c_{ij} \Delta x_j + \sum_{k=1}^{n_g} d_{ik} \Delta g_k, \quad i = \overline{1, m} \qquad (1)$$

$$\Delta x_i = \sum_{v=1}^{n_v} p_{jv} \Delta v_v + \sum_{\mu=1}^{n_\mu} r_{j\mu} \Delta u_\mu, \; j = \overline{1, n_x},$$

где q_i, c_{ij}, d_{ik}, p_{jv}, и $r_{j\mu}$ — коэффициенты линейной множественной регрессии между соответствующими переменными.

Система уравнений (1) может служить основой структурно-параметрического моделирования технологических систем и описывать их аномальные состояния. При этом все контролируемые параметры целесообразно привести к безразмерной шкале относительных величин

$$z_{ij} = \frac{z_{ij}^t - z_{ij}^0}{\Delta z_{ij}^0},$$

где z_{ij} — j-й параметр i-го множества G, U, V, X и Y; z_{ij}^t и z_{ij}^0 — фактическое и нормативное значения j-го параметра в i-м множестве; Δz_{ij}^0 — предельно допустимое отклонение от нормы.

Для сгруппированного множества наблюдаемых факторов критерий оценки состояния системы можно записать в виде аддитивно-мультипликативной свертки [3]

$$Q = \prod_{k=1}^{m_k} (1 - z_k^2) \cdot \left[\sum_{i=1}^{m} a_i \left(1 - \sqrt{\sum_{j=1}^{n_j} b_{ij} z_{ij}^2} \right) \right], \qquad (2)$$

где z_{ij} и b_{ij} — относительное отклонение j-го фактора i-й группы и его весовой коэффициент; a_i — коэффициент значимости i-й группы факторов; z_k — отклонение k-го фактора критической группы, однозначно определяющей неприемлемость качественного состояния процесса или продукта.

При $\sum_{i=1}^{m} a_i = 1$ и $\sum_{i=1}^{m} b_{ij} = 1$, $i = \overline{1, m}$ функционал (2) изменяется от 1 до 0, соответственно, от эталонного состояния до его граничного допустимого значения и обращается в нуль при выходе любого

параметра критической группы за предельно допустимый уровень.

Для оценки групповых и индивидуальных весовых коэффициентов формируется таблица экспертных оценок функционала по схеме полного или дробного факторного эксперимента с последующим вычислением оценок линейных эффектов влияния на функционал по формулам

$$a_i = \frac{1}{N}\sum_{k=1}^{N} z_{ik} Q_k$$

$$b_{ij} = \frac{1}{N}\sum_{k=1}^{N} z_{ijk} Q_k,\ j = \overline{1, n_i},\ i = \overline{1, m},$$

где z_{ik} — значение i-го группового показателя в k-м опыте; z_{ijk} — значение j-й переменной i-й группы в k-м эксперименте; Q_k — значение функционала в k-м эксперименте.

Уравнения (1), описывающие структуру и значимость прямых связей между контролируемыми факторами, позволяют перейти к матричным описаниям функциональной структуры системы путем упорядочивания множеств отклонений Q, Y, X, G, V и U в виде квазидиагональной матрицы

$$\begin{Vmatrix} \Delta Q & & & & & \\ & \|\Delta y_i \delta_{ij}\|^m & & & & \\ & & \|\Delta x_j \delta_{jk}\|^{n_x} & & & \\ & & & \|\Delta g_k \delta_{kl}\|^{n_g} & & \\ & & & & \|\Delta v_v \delta_{vx}\|^{n_v} & \\ & & & & & \|\Delta u_\mu \delta_{j\mu}\|^{n_u} \end{Vmatrix} \quad (3)$$

(δ — символ Кронекера).

В соответствии с этим описанием можно составить клеточную структурную матрицу, систематизирующую по отдельным блокам совокупность всех принципиально возможных матриц характеристик взаимосвязей между элементами параметрических групп:

$\|1\|$ 01	$\|q_i\|$ 02	03	04	05	06
$\|0\|$ 07	$\|\delta_{ij}\|^m$ 08	$\|c_{ij}\|$ 09	$\|d_{ij}\|$ 10	11	$\vdots$ 12
$\|0\|$ 13	$\|0\|$ 14	$\|\delta_{jk}\|^{n_x}$ 15	16	$\|p_{jv}\|$ 17	$\|r_{j\mu}\|$ 18
$\|0\|$ 19	$\|0\|$ 20	$\|0\|$ 21	$\|d_{kl}\|^{n_g}$ 22	23	24
$\|0\|$ 25	$\|0\|$ 26	$\|0\|$ 27	$\|0\|$ 28	$\|d_{v\tau}\|^{n_v}$ 29	30
$\|0\|$ 31	$\|0\|$ 32	$\|0\|$ 33	$\|0\|$ 31	$\|0\|$ 35	$\|\delta_{\mu\xi}\|^{n_u}$ 36

(4)

где $\|q_i\|$, $\|c_{ij}\|$, $\|d_{ik}\|$, $\|p_{jv}\|$, $\|r_{j\mu}\|$ — матрицы коэффициентов связей между векторами соответствующих переменных.

Квадраты главной диагонали структурной матрицы (4) объединяют операторы функциональных связей внутри выделенных групп параметров и в случае их независимости выражаются единичными диагональными матрицами $\|\delta_{ij}\|^m$, $\|\delta_{jk}\|^n$ и т. д. В случае взаимосвязанности элементов внутри группы в соответствующую клетку помещается матрица оператора взаимодействия

$$\|\varphi_{jk}\|^n = \begin{Vmatrix} 1 & \varphi_{12} & \dots & \varphi_{1n} \\ \varphi_{21} & 1 & \dots & \varphi_{2n} \\ \dots & \dots & \dots & \dots \\ \varphi_{n1} & \varphi_{n2} & \dots & 1 \end{Vmatrix}, \quad (5)$$

где φ_{jk} — нормированные коэффициенты или функции связи между j-м и k-м элементами в системе уравнений

$$\Delta x_j = \sum_{k=1}^{n} \varphi_{jk} \Delta x_k,\ j = \overline{1, n},\ k \neq j.$$

В общем случае функция связи φ_{ij} отражает интенсивность влияния j-го параметра на i-й и каждая строка матрицы описывает вектор входных (причинных) связей, влияющих на i-й показатель состояния системы. В свою очередь, каждый j-й столбец матрицы описывает вектор следственных связей j-го фактора с другими параметрами состояния.

Недиагональные клетки матрицы (4) соответствуют операторам прямого и косвенного влияний различных функциональных групп друг на друга и на показатели целевой функции или функционал. При этом пустые клетки означают возможные, но неизвестные операторы взаимосвязей, а нулевые матрицы $\|0\|$ априори определяют область несуществующих связей.

Таким образом, клеточная матрица (4) представляет собой полное описание структуры и характеристик связей между параметрами и факторами, определяющими функционирование любого технологического процесса или комплекса. Детализация параметрических описаний объекта в виде клеточной матрицы определяется конкретной задачей идентификации и моделирования системы, а также характером априорных данных о качественных и количественных показателях влияния и взаимодействия между ее элементами.

Коэффициенты связей в виде сопоставимых количественных характеристик могут быть найдены известными методами факторного анализа, планирования эксперимента и экспертных оценок в зависимости от степени априорных представлений о природе связей.

При имеющихся статистических данных о состоянии системы и среды в виде массива x_{kj}, $k = \overline{1, m}$, $j = \overline{1, n}$, где x_{kj} — значение j-го фактора в k-м опыте, составляется матрица коэффициентов корреляции

$$r_{ij} = \frac{1}{m-1} \cdot \sum_{k=1}^{m} \frac{(x_{ki} - \bar{x}_i)}{S_{x_i}} \cdot \frac{(x_{kj} - \bar{x}_j)}{S_{x_j}},$$

где $\bar{x}_i$ и $\bar{x}_j$ — средние значения i-го и j-го факторов; S_{x_i}, S_{x_j} — среднеквадратичные отклонения соответствующих факторов.

Характер связей между коррелируемыми факторами определяется коэффициентами линейной множественной регрессии

$$\Delta x_i = \sum_{j=1}^{n_i} P_{ij} \Delta x_j, \quad j = \overline{1, n}$$

с коэффициентами связи P_{ij} j-го фактора с i-м.

Для сопоставимой оценки отклонений и связей параметров различной физической природы и разной размерности формируется матрица безразмерных характеристик

$$C_{ij} = \frac{P_{ij} \Delta x_j^0}{\Delta x_i^0}, \; i, j = \overline{1, n},$$

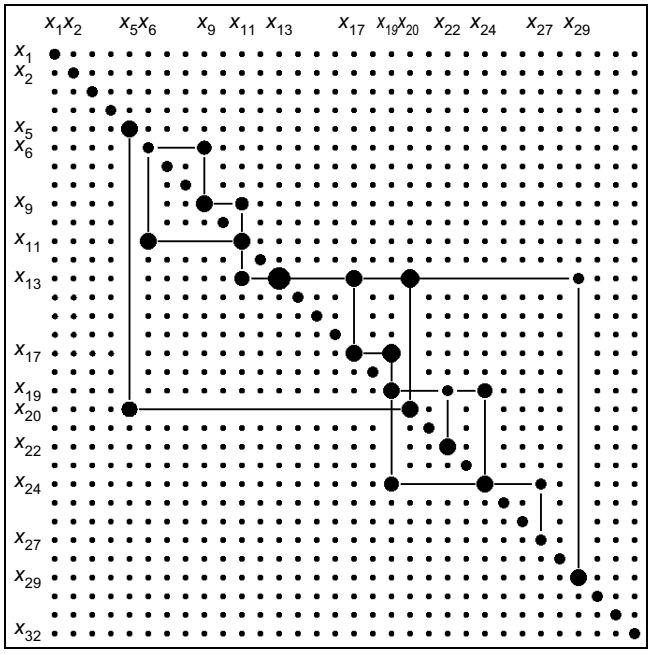
где Δx_i^0 и Δx_j^0 — допустимые отклонения от нормы.

По найденной матрице характеристик взаимосвязей C_{ij}, $i, j = \overline{1, n}$, и вектору текущих отклонений $\Delta x_1, \ldots, \Delta x_n$ путем умножения матрицы $\|C_{ij}\|^n$ на диагональную матрицу вектора изменений параметров состояния $\|\Delta x_j \delta_{jk}\|$ формируется *ситуационная матрица*

$$\begin{Vmatrix} 1 & c_{12} & \ldots & c_{1n} \\ c_{21} & 1 & \ldots & c_{2n} \\ \ldots & \ldots & \ldots & \ldots \\ c_{n1} & c_{n2} & \ldots & 1 \end{Vmatrix} \cdot \begin{Vmatrix} \Delta x_1 & & & \\ & \Delta x_2 & & \\ & \ldots & \ldots & \ldots \\ & & & \Delta x_n \end{Vmatrix} =$$

$$= \begin{Vmatrix} \Delta x_1 & c_{12}\Delta x_2 & \ldots & c_{1n}\Delta x_n \\ c_{21}\Delta x_1 & \Delta x_2 & \ldots & c_{2n}\Delta x_n \\ \ldots & \ldots & \ldots & \ldots \\ c_{n1}\Delta x_1 & c_{n2}\Delta x_2 & \ldots & \Delta x_n \end{Vmatrix}, \qquad (6)$$

описывающая разложение отклонений Δx_i, $i = \overline{1, n}$, по всем координатам множества $\{X\}$, объединяя, таким образом, априорные данные о структуре связей с текущей информацией Δx. При этом элементы главной диагонали ситуационной матрицы (6) отображают текущие отклонения Δx_i контролируемых факторов от заданных значений, а недиагональные — вклады отклонений Δx_j, $j = \overline{1, n}$, в отклонения Δx_i, $i = \overline{1, n}$, с упорядочением по строкам всех априори известных причин отклонений Δx_i, а по столбцам — возможных следственных влияний отклонений Δx_i на другие параметры.



Структурно-параметрическая ситуационная модель аномального состояния в пространстве контролируемых параметров $x_1, \ldots, x_{32}$

Ситуационная матрица (6) позволяет определить формальную *процедуру идентификации аномальной ситуации*, представленной на рисунке в качестве примера в аналоговой форме. Исходя от максимального диагонального элемента, соответствующего максимальному отклонению от нормы — Δx_{13} в наблюдаемом множестве параметров состояния системы, следует перемещение по строке (см. рисунок) с выявлением причин, вызвавших отклонение данного параметра, и выбором наиболее значимой. Далее следует переход по столбцу к новому элементу главной диагонали, после чего вновь оцениваются элементы соответствующей строки. Поиск продолжается до нахождения отклонения, в строке которого все недиагональные элементы будут равны нулю, например Δx_5. Это означает, что данное отклонение — одна из основных исходных причин возникновения аномальной ситуации.

Алгоритм идентификации содержит блок формирования ситуационной матрицы и процедуру поиска причин аномального состояния системы. Процедура поиска представляет собой цикл пере-

бора независимых отклонений, внутри которого происходит отыскание максимального элемента в строке, запоминание его порядкового номера r и переход на r-ю строку с повторением поиска максимального элемента этой строки. Для обнаружения возможного зацикливания причинно-следственных связей формируется массив индексов диагональных элементов, входящих в траекторию взаимодействия. Совпадение двух элементов этого массива служит признаком возникновения цикла. При этом причина может оказаться в контуре цикла или вне его. Для выхода из цикла и продолжения поиска исходной причины разрывается последнее звено обратной функциональной связи и при повторном переборе элементов предшествующей строки процедура либо остановится на последней вершине цикла (если причина находится в контуре цикла), либо пойдет дальше по ступеням взаимосвязей до следующей причины или нового цикла (см. рисунок).

При переходе к нахождению других траекторий воздействия на исследуемое отклонение максимальный вклад в него приравнивается к нулю и выбирается следующий по размеру вклад (т. е. следующий максимальный элемент строки).

Ситуационная матрица (6) дает возможность прогнозировать развитие ситуации по выходным параметрам состояния (по параметрам качества) при отклонении от норм определенных показателей состояния системы, технологических режимов и управляющих воздействий.

Алгоритм прогноза связан с имитацией отклонения какого-либо фактора Δx_k от нормативного значения, вычислением элементов k-го столбца ситуационной матрицы как $S_{ik} = c_{ik}\Delta x_k$, $i = \overline{1, n}$, и нахождением в нем максимального недиагонального элемента S_{qk}, соответствующего максимальному следственному воздействию $c_{qk}\Delta x_k = \max$ на q-й параметр. Если оно оказывается равным нулю, то k-е отклонение не имеет последствий в контролируемом n-факторном пространстве и после регистрации индексного массива причинно-следственных отклонений процедура заканчивается. При $\max \neq 0$ индекс следственного отклонения q записывается в очередной элемент индексного массива и после проверки на зацикливание следует вычисление отклонения Δx_q (диагонального элемента q-й строки матрицы S_{ij}) как $\Delta x_q = c_{qk}\Delta x_k$ с дальнейшим повторением процедуры в k-м столбце при $k = q$ и $\Delta x_k = \Delta x_q$. Обнаружение причинно-следственных циклов аналогично алгоритму идентификации аномальных ситуаций.

Для определения всех ветвей прогнозируемого аномального состояния системы описанная процедура включается в алгоритм их последовательного перебора по принципу разматывания и сматывания нити в конечном лабиринте. При достижении тупикового элемента очередной ветви последнее ее звено разрывается, т. е. соответствующая связь принимается равной нулю с возвратом к предшествующей ступени (сматывание нити) и нахождением следующего наибольшего вклада k-го элемента, т. е. другой ветви причинно-следственного воздействия (разматывание нити). Процедура останавливается при достижении исходного пункта.

В случае нескольких входных отклонений, т. е. некоторого вектора Δx_j, $i = \overline{1, n}$, формируется матрица прогнозируемой ситуации с запуском процедуры прогнозирования последовательно для всех исходных отклонений (истоков аномалии).

Составление структурно-параметрических моделей (3)—(5) связано с построением структурной матрицы большой системы [4] и сводится к следующим этапам:

- представление системы как элемента инфраструктуры с определением внешних связей и факторов, влияющих на ее внутреннюю структуру;
- разработка для собственно технологической системы крупноблочной матрицы (3), каждый блок которой соответствует либо определенному функциональному участку, либо элементу декомпозиции главной целевой функции; выделенные блоки ориентируются вдоль главной диагонали по ходу процесса либо по целевому или функциональному принципу;
- детализация элементов крупноблочной матрицы с разделением каждого диагонального блока матрицы на составные элементы — технологические операции, определяющие факторы и параметры описания функциональных блоков;
- нахождение характеристик связей между выделенными факторами и группами факторов методами экспертных оценок и факторного анализа.

Рассмотренные алгоритмы идентификации состояния больших систем на основе структурно-параметрических моделей позволяют решать задачи принятия решений в сложной ситуации при нечетких аналитических описаниях функциональных связей, основанных лишь на оценках влияния и отношений порядка. Процедура прогноза лежит в основе работы *модуля принятия решения* и определения вектора изменения параметров состояния системы в направлении улучшения целевой функции. Алгоритм принятия решений сводится к накоплению базы знаний и составлению структурно-параметрической модели связей между параметрами состояния и целевой функции и последующим процедурам причинно-следственной идентификации и прогнозирования текущей ситуации на очередном шаге движения к цели.

Описанная информационная технология может быть полезной в различных сферах деятельности (производственной, экономической, социальной, экологической и др.), поскольку она позволяет решать две основные задачи анализа и принятия решений в структурно-сложной ситуации, а именно:

- что в поведении системы, среды или других активных элементов явилось причиной аномалии? (задача идентификации);
- что произойдет с системой при изменении тех или иных параметров ее состояния или окружающей среды? (задача прогнозирования).

Рассмотренные задачи и алгоритмы их решения положены в основу процедур поддержки принятия решений по управлению качеством готовой продукции в технологической системе перерабатывающего предприятия агропромышленного комплекса [5].

ЛИТЕРАТУРА

1. *Ивашкин Ю. А.* Структурно-параметрические модели и алгоритмы идентификации аномальных состояний технологической системы // Вестн. Росс. акад. диалект.-систем. исслед. — 1998. — Вып. 2. — С. 18—28.
2. *Ивашкин Ю. А.* Матричный метод отображения оперативной информации //Приборы и системы управления. — 1970. — № 10. — С. 15—16.
3. *Ивашкин Ю. А.* Компьютерные технологии оптимальных решений в переработке биосырья // Докл. Третьей междунар. науч.-техн. конф. “Пища. Экология. Человек”, — М., 1999. — С. 99—105.
4. *Шатихин Л. Г.* Структурные матрицы и их применение для исследования систем. — М.: Машиностроение, 1991.
5. *Gordeeva Y. L., Ivashkin Y. A.* The informational technology of the quality control and the security of products // CHISA, 2002. — Praha, 2002.

☎ *(095) 277-07-30*

E-mail: ivashkin@msaab.ru

❑

УДК 681.325...2

ПРИНЦИПЫ ПОСТРОЕНИЯ ИНТЕЛЛЕКТУАЛЬНОГО ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ ДЛЯ СИСТЕМ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ОПЕРАТОРОМ

В. Г. Лебедев

Институт проблем управления им. В. А. Трапезникова, г. Москва

Рассмотрена архитектура верхнего уровня интеллектуального интерфейса оператора, управляющего сложным аппаратно-программным комплексом, и функциональные особенности его основных модулей.

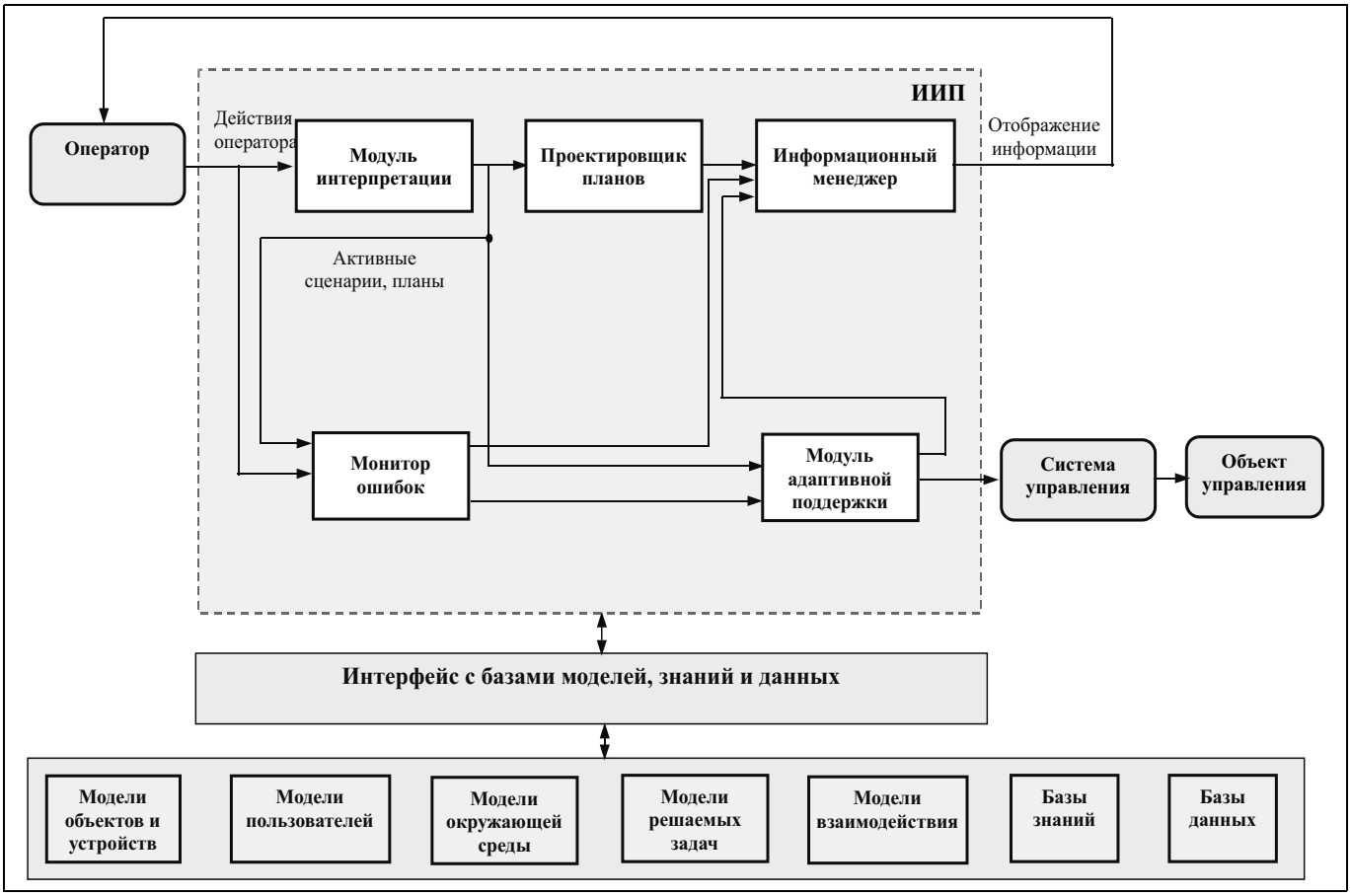
Будем рассматривать интеллектуальный интерфейс пользователя (ИИП) как составную часть системы поддержки принятия решений оператором (кратко — СПО), предназначенной для оказания помощи оператору (пользователю) в преодолении возникающих проблем при его взаимодействии со сложным аппаратно-программным комплексом путем расширения возможностей данного интерфейса по сравнению с традиционным графическим интерфейсом пользователя.

Эти возможности должны обеспечивать адаптивное к пользователю и решаемой задаче взаимодействие оператора с системой, диалоги между пользователем и системой, представление информации в интегрированном исчерпывающем виде.

Интеллектуальный интерфейс пользователя можно считать ключевым модулем СПО поскольку он отвечает за выбор и передачу информации от других модулей для её отображения, за задержку или удаление информации, а также за выполняющиеся действия от имени оператора, так как поддерживает собственные цели оператора как лица, принимающего решения.

Входная информация, модели и знания используются в ИИП для выяснения намерений оператора, и ответов, в частности, на вопросы:

- какую информацию необходимо представить оператору?
- какие задачи должны выполняться автоматически?



Архитектура верхнего уровня интеллектуального интерфейса пользователя

- какие действия или бездействие оператора ошибочны?

Для решения упомянутых задач предлагается структура ИИП, включающая в себя:

- *модуль интерпретации*, трактующий текущие планы оператора для описания его действий и поддержания активного графа целей, как модели намерений оператора;
- *монитор ошибок*, определяющий, имеют ли действия или бездействие оператора серьезные отрицательные последствия в контексте существующей ситуации;
- *модуль адаптивной поддержки*, использующий вектор состояния объекта управления (ОУ), и планы действий для определения задач, которые должны быть выполнены, и передачи соответствующих команд ОУ;
- *проектировщик планов*, объединяющий сформированные планы в предложения для оператора через информационного менеджера;
- *информационный менеджер*, представляющий предложения по планам от проектировщика планов, управляющий отображением информации и дающий оценки планов оператору в необходимое время и необходимым образом.

На рисунке представлена архитектура верхнего уровня ИИП. Рассмотрим функциональные особенности его модулей.

МОДУЛЬ ИНТЕРПРЕТАЦИИ

В этом модуле действия оператора (команды для изменения состояния ОУ) интерпретируются в терминах уровня целей — желательных состояниях ОУ или ситуаций и уровня планов − методов изменения состояния ОУ для получения желательного состояния.

Фактически все взаимосвязи между оператором и системой в современных СПО находятся на низком уровне абстракции — обычно через средства управления. Модуль интерпретации приводит к взаимодействию на более высоком уровне. Преимущества заключаются в том, что планы и цели оператора служат входом для моделей СПО, инициирующей соответствующие действия.

Модуль интерпретации является новой формой связи между оператором и ОУ. Единственное явное выражение намерений оператора в ИИП — через принятие или отклонение предложенных планов от проектировщика планов. Все другие выра-

жения намерений оператора поступают в ОУ неявно. Это не приводит ни к какой дополнительной нагрузке на оператора (т. е. оператор не сообщает своих намерений) и не вызывает никаких ограничений на действия оператора.

Неявная связь с проектировщиком планов также поддерживается с помощью интерпретации цели. В системах реального времени оператору легче выполнить шаги плана, сгенерированные системой в результате интерпретации цели. Результаты процесса интерпретации приводят к активизации планов и целей в модели цели, которые передаются оператору проектировщиком планов. Если оператор принимает предложение, активизируются планы и цели, которые были предложены проектировщиком планов. Оператор отвергает план, в частности, когда планы и цели, которые должны быть активизированы, взаимоисключающие с теми, которые в настоящее время преследуются. В любом случае об отклонении сообщается другим модулям ИИП.

Возможны две разновидности интерпретации — основанная на сценарии, которая используется для достаточно хорошо определённой последовательности действий, и основанная на плане, которая используется для менее структурированных действий.

Преимущество сценарной интерпретации заключается, прежде всего, в быстроте, так как не требуется никакого поиска (в отличие от интерпретации, основанной на плане). В процессе рассуждения, основанном на сценарии, используется набор предварительно заполненных процедурных представлений, называемых сценариями, чтобы найти объяснения действиям оператора. Сценарии представляют собой стереотипное поведение в данной ситуации.

Рассуждение, основанное на плане, используется для объяснения действий оператора при достижении одной из его целей. Для формирования заключения о намерениях используется база знаний, которая включает в себя два типа знаний: интерпретация и взаимное исключение.

Интерпретация знаний описывает условия, при которых дочерний план (или цель) может интерпретироваться как часть плана (или цели) родителя. Если гипотетические планы и цели не могут быть разработаны, то осуществляется обратный поиск и проверка другого набора. В противном случае гипотетические планы и цели становятся активными.

Далее используются знания о взаимном исключении, чтобы дезактивировать любые планы и цели, которые противоречивы по отношению к только что активизированным планам и целям.

Сценарии, планы и цели широко применяются в СПО. В рамках ИИП они используются модулем адаптивной поддержки для помощи при решении задач управления объектом, информационным менеджером для управления отображением, монитором ошибок для поиска контекста.

МОНИТОР ОШИБОК

Данный модуль выявляет ошибки оператора с серьезными последствиями. Если такая ошибка обнаружена, монитор ошибок пытается сообщить об этом оператору, запрашивает модуль адаптивной поддержки, чтобы он выполнил сценарий корректировки ошибки оператора. Для автоматической корректировки ошибки монитор ошибок должен быть уполномочен на это оператором.

Цель монитора ошибок состоит в предотвращении последствий ошибки оператора, которая может привести к значительному ущербу, неудачной операции, потере ОУ и т. п. Он предназначен именно для предотвращения серьезных последствий ошибок, а не непосредственно ошибок. Его работа состоит их двух фаз.

На первой фазе монитор ошибок использует схему классификации ошибок, основанную на упущениях, повторениях, ошибочных операциях. Например, при выполнении последовательности действий операторы иногда пропускают, или повторяют шаги или выполняют шаги, не входящие в последовательность. Как только обнаружена ошибка этого типа, предсказываются последствия ошибочного действия.

На этой фазе монитор ошибок:

- может определить упущения, повторения или ошибки в последовательности выполнения сценариев;
- обеспечивает помощь после совершенного ошибочного действия с постепенным наращиванием предупреждений (поскольку ОУ переходит ближе к опасному состоянию);
- может обнаружить только процедурные ошибки вида, найденного в контрольных списках.

На второй фазе монитор ошибок отслеживает действия оператора и изменяет состояние в сети ошибок. Каждая сеть ошибок — конечный автомат, предназначенный для обнаружения некоторого типа серьезной ошибки. Действия оператора или изменение состояния ОУ могут привести к переходу от одного состояния в сети к другому. Когда ОУ перемещается ближе к опасному состоянию, повышается уровень реакции от общего отображения и предупреждения (через информационного менеджера) до управления (через адаптивную поддержку).

Действия оператора подвергаются последовательному анализу и, если не существует никакой опасной последовательности, то не производится никакой дальнейшей обработки действий с помощью монитора ошибок. Ошибочные действия в дальнейшем анализируются монитором ошибок, который пытается классифицировать упущения,

повторения, ошибки в последовательности шагов сценария. Эта классификация выполняется путем согласования ошибочных действий с шагами, которые уже завершены в каждом активном сценарии.

Монитор ошибок предсказывает действия оператора и определяет серьезность последствий для текущей ситуации и ситуации ближайшего времени.

Существует три уровня посредничества при контроле ошибок.

Расширение отображения, заставляющее оператора обнаруживать развивающуюся ошибку.

Предупреждение. Когда оператор не распознает ошибки и продолжает путь к серьезной ошибке, то его внимание может быть привлечено к ситуации путем изменения отображения с помощью различных типов представления (акустических, визуальных).

Управление. Когда предпринятые шаги не приводят к удаче или нет времени для любой другой постепенной меры посредничества, то для избежания возможной катастрофы монитор ошибок предлагает через модуль адаптивной поддержки осуществить управление объектом.

МОДУЛЬ АДАПТИВНОЙ ПОДДЕРЖКИ

Адаптивная поддержка решает задачи автоматически по поручению оператора. Она помогает оператору, если он нуждается в поддержке при выполнении эксплуатационных требований, которые могут возникнуть в процессе управления.

Автоматизация большого количества технологически возможных задач имеет некоторые нежелательные для оператора побочные эффекты: оператор должен контролировать автоматику, ролью которой не всегда удовлетворен; когда оператор берет управление на себя, то существует период слабого управления объектом, пока оператор не привыкнет к действиям системы; автоматика может действовать непредсказуемо в некоторых состояниях ОУ и, таким образом, быть неуклюжей при управлении.

Адаптивная поддержка помогает оператору в следующих ситуациях:

- оператор слишком занят, чтобы выполнять ещё одну задачу;
- оператор желает автоматически выполнить некоторую задачу всякий раз, когда она уместна (т.е. когда она предлагается проектировщиком плана);
- оператор не способен управлять ОУ самостоятельно (по различным причинам);
- оператор совершает ошибку, которую сам не обнаруживает;
- угрожающая ситуация требует немедленного ответа.

Адаптивная поддержка может обеспечивать три формы помощи: преобразование, разделение и размещение.

Задача *преобразуется*, чтобы упростить ее для выполнения оператором. Например, модуль адаптивной поддержки может изменить отображение, чтобы высветить оптимальный рабочий диапазон некоторого параметра ОУ для помощи оператору при управлении этим параметром.

Задача может быть *разделена* для совместного решения с оператором. Здесь предполагается максимальное взаимодействие СПО с оператором.

Наконец, модуль адаптивной поддержки может *размещать* задачу для автоматического выполнения.

Решение любой задачи с помощью адаптивной поддержки требует модели взаимодействия с оператором, способствующей повышению качества принимаемых решений.

ПРОЕКТИРОВЩИК ПЛАНОВ

Данный модуль служит для расположения планов по приоритетам и представления предложений планов оператору. В нем фильтруются различные входящие предложения и представляются оператору. Он также отмечает принятые и отвергнутые предложения.

Проектировщик планов посылает предложения информационному менеджеру, принимая во внимание выбор времени, уместность по отношению к другим действиям и пр. Предложения посылаются упорядоченным образом, связывая планы общей целью и с планом более высокого уровня.

Если оператор принимает предложение, проектировщик планов соединяется со всеми связанными подсистемами, чтобы они могли выполнить необходимые функции.

Проектировщик планов объединяет отдельные тактики, поручения и планы в последовательные предложения от информационного менеджера к оператору. Такое объединение очень важно для оператора, так как трудно понять важность и значение плана, если он не представлен с достаточным контекстом. Проектировщик планов собирает различные аспекты предложений плана вместе и представляет их оператору как одну идею.

Проектировщик планов предполагает, что оператор занимается планированием параллельно с ним. Поэтому окончательная цель проектировщика планов заключается в представлении планов, которые отражают собственные планы оператора, но более уточненные по сравнению с ними, возможно, оптимальные.

Проектировщик планов синтезирует поток планов, каждый из которых должен быть принят или отклонен. Атрибуты плана содержат внутреннее имя плана (например, используемая ракета) и не-

которую уместную информацию относительно плана (например, против объекта *N*). Планы могут быть групповые, учитывающие, например, группы летательных аппаратов, их маневры, вооружение и др.

План выбирается с помощью базы знаний, которая ведет проектировщика планов. Например, проектировщик планов должен иметь глубокие знания о механизмах управления, знания о семантических взаимосвязях среди планов и др.

Оператору необходимо, чтобы проектировщик планов предложил планы следующим образом:

- своевременно;
- с соответствующей скоростью, которая может быть воспринята;
- в форме, которая может быть с готовностью понята;
- в интерактивном режиме, но предоставляя оператору управление;
- способом, который позволяет исследовать альтернативы.

Это поведение на верхнем уровне транслируется в следующие требования к проектировщику планов:

- обеспечить интерфейс, основанный на графе "план — цель", через который планировщик сможет выдавать предложения;
- выдвигать предложения информационному менеджеру с характеристиками, понятными оператору;
- позволять критическим предложениям выгружать некритические предложения;
- обеспечить механизм принятия и отклонения предложений, поддерживающий как явные (через оператора), так и неявные (через вывод заключений для поддержки оператора) принятие и отклонение;
- сообщать другим блокам СПО о принятии или отклонении предложений;
- обеспечить механизм, позволяющий оператору исследовать альтернативы для начальных и модифицированных предложений (например, маршрут нарушений) до их принятия.

Проектировщик планов служит интерфейсом между модулем интерпретации и информационным менеджером. Сгенерированные планы представляются оператору, который принимает или отвергает их явно; в частности, предложения отвергаются, если их реализация не укладывается по времени.

ИНФОРМАЦИОННЫЙ МЕНЕДЖЕР

Преимущество автоматизированного информационного менеджера заключается в том, что оператор может явно не управлять отображением для анализа информации. Информационный менеджер расширяет ситуационное понимание оператора, предъявляя более эффективные решения оператору с требуемой информацией. Это поддерживает оператора, получающего своевременную разумную информацию в легко интерпретируемом виде. При этом информация, не соответствующая планам ближайшего времени и текущей ситуации, не отображается.

Информационный менеджер:

- всегда показывает, что требует оператор;
- сокращает нагрузку на оператора, конфигурируя средства управления отображением автоматически;
- обеспечивает оператора информацией, которая ему необходима для завершения планов;
- подготавливает оператора к будущим событиям;
- представляет сформированные планы оператору.

Как только информационные требования становятся полностью известными, информационному менеджеру необходимо выбрать средства отображения, чтобы выполнить эти требования. Выбор отображений является задачей планирования по распределению и расположению. Распределение заключается в определении размера пространства для каждой части информации и предполагает определение позиции расположения информации относительно некоторых устройств отображения. В вычислительном отношении решение проблемы распределения и расположения требует больших затрат, так как много возможных конфигураций необходимо исследовать для поиска оптимальной конфигурации.

Информационный менеджер состоит из двух блоков: сборщика требований и распределителя устройств отображения. Первый из них собирает информационные требования от источников (планов и событий) и сортирует их по важности. Одинаковые требования, относящиеся к одной и той же информации, объединяются. Второй анализирует информационные требования в текущий момент времени и распределяет ресурсы средств отображения. Данный блок принимает ряд решений: что отображать, какую информацию выделить, какие средства отображения использовать (аудио, видео, текст, графика), какие использовать атрибуты (цвет, размер, интонация) и т. д.

Информационный менеджер — один из наиболее важных модулей в ИИП. Это связано с ответственностью за любую ошибку в выборе средства отображения. Точное рассмотрение выбора средств отображения — довольно трудная задача. Проектировать информационный менеджер легче, когда оператор не вмешивается в процесс отображения информации. Полностью автоматическое формирование информации проще, но необходимо помнить, что оператор должен всегда иметь возможность вмешаться в работу СПО.

☎ *(095) 334-92-31*
E-mail: valya@ipu.ru

❑

УДК 681.518.54;004.3.001.4

САМОДИАГНОСТИРОВАНИЕ ВОЗНИКАЮЩИХ ОТКАЗОВ И ИХ УСТРАНЕНИЕ В ЦИФРОВЫХ СИСТЕМАХ С РЕКОНФИГУРАЦИЕЙ

В. А. Ведешенков

Институт проблем управления им. В. А. Трапезникова, г. Москва

Предложен метод самодиагностирования (СД) и устранения возникающих отказов компонентов (модулей и линий связи) цифровых систем с реконфигурацией. Допускаются устойчивые кратные отказы ограниченного числа компонентов, причем в процессе диагностирования может возникнуть одиночный отказ. Для обнаружения таких отказов и выработки правильного диагноза осуществляется дву- и более кратный запуск процесса СД, результаты двух последних запусков сравниваются. Для устранения отказавших компонентов путем смены рабочих конфигураций применяется вариант скользящего резервирования. Дан пример реализации СД одиночного отказа, возникшего в процессе СД, и реконфигурации 2-отказоустойчивой системы.

ВВЕДЕНИЕ

Многопроцессорные вычислительные системы (МВС) широко применяются в автоматизированных системах управления различными объектами. Необходимая отказоустойчивость МВС часто достигается с помощью стратегии динамической избыточности, включающей в себя этапы контроля работоспособности системы, идентификации отказавшего компонента, реконфигурации для его замены исправным компонентом из числа резервных, восстановления функционирования системы [1, 2]. Методы синтеза графа структуры отказоустойчивых систем, предложенные в работе [2], позволяют построить граф с заданной степенью m отказоустойчивости (m-ОУ граф) для целевого (исходного неизбыточного) графа достаточно общего вида. Применение инвариантно-группового подхода приводит к получению структур m-ОУ графов регулярного вида, что упрощает процедуры их диагностирования и реконфигурации. Но работ по диагностированию отказавших компонентов и реконфигурации в отказоустойчивых системах немного [3—6], причем в работе [6] основное внимание было уделено разработке метода СД устойчивых отказов компонентов цифровых систем с реконфигурацией.

В настоящей статье при тех же предпосылках об объекте исследуются особенности СД тех компонентов, которые отказывают во время процесса СД. Результаты взаимного тестирования отдельных частей системы (модулей), как и в работе [6], описываются с помощью модели Барси—Грандони—Маестрини (сокращенно, модели БГМ) [7]. Суммарный результат проверки модулей системы образует двоичный вектор R, который называется реальным синдромом [7, 8]. Поскольку в данной работе тестовая информация будет представляться в табличном виде, то в дальнейшем будем говорить об обработке табличной информации.

Ниже представлен метод СД возникающих отказов цифровых систем с реконфигурацией, опирающийся на принцип расширяющихся областей, который состоит в том, что выполнение необходимых процедур начинается с исправного модуля и последовательно передается другим модулям, исправность которых подтверждена предшествующими проверками. Основой применения этого принципа служит следующая гипотеза модели БГМ: нулевой результат, выдаваемый проверяющим модулем (МПщ), подтверждает исправность проверенного им модуля, поэтому для идентификации исправных модулей достаточно 0-путей единичной длины (в терминологии работы [9]). Добавим, что начальный модуль СД не является фиксированным, а определяется по результатам работы системы контроля [10].

СВЕДЕНИЯ ОБ ОБЪЕКТЕ ДИАГНОСТИРОВАНИЯ

Структура неизбыточной МВС представляется графом G_0^n, вершинам которого соответствуют однородные группы ресурсов (процессоры, модули), а дугам — линии связи между ресурсами. Предполагается, что отказывать могут как вершины, так и дуги, причем отказ вершины приводит к отказу всех инцидентных ей (входящих и выходящих) дуг. Алгоритмы синтеза [2] позволяют построить для n-вершинного целевого графа G_0^n и заданной степени m отказоустойчивости искомый объемлющий граф G_R^{n+m}. Наличие гамильтонова цикла в целевом графе G_0^n является необходимым условием существования отказоустойчивого графа G_R^{n+m}.

Граф G_R^{n+m} m-отказоустойчивой (ОУ) системы содержит $(n+m)$ вершин, из которых n вершин соответствуют рабочим модулям, а оставшиеся m вершин — резервные. Он симметричен относительно $(n+m)$ поворотов (сдвигов) вокруг центра графа на угол $dg_1 = 360^\circ/(n+m)$, т. е. при повороте на угол dg_1 каждая вершина вместе с инцидентными ей дугами совмещается с соседней вершиной и инцидентными ей дугами. Такая симметрия графа G_R^{n+m} обеспечивает замену отказавшей вершины и инцидентных ей дуг на исправную вершину и инцидентные ей дуги путем поворота на соответствующий угол, кратный dg_1.

Обозначим символом $c_v(c_a)$ максимальное число отказавших вершин (внешних дуг) m-ОУ графа G_R^{n+m}, которые могут быть исключены из состава работающей системы путем смены рабочей конфигурации (РК); тогда справедливо соотношение

$$c_v + c_a \le m. \qquad (1)$$

Соотношение $c_v \le m$ при $c_a = 0$ очевидно, так как, по определению, граф G_R^{n+m} является m-ОУ относительно отказов m его вершин. Для устранения отказавшей внешней дуги, соединяющей соседние вершины гамильтонова цикла, необходимо вывести из состава РК (и не использовать в последующих РК) хотя бы одну из этих вершин, т. е., по существу, считать ее отказавшей. Таким образом, для устранения c_a отказавших внешних дуг нужно исключить c_a вершин, если эти дуги не имеют общих вершин. Неравенство в соотношении (1) учитывает влияние возможных отказов внутренних дуг, которые могут понизить оценки для чисел c_v и c_a.

Для примера на рис. 1 показан исходный целевой граф G_0^8: внешние дуги образуют гамильтонов

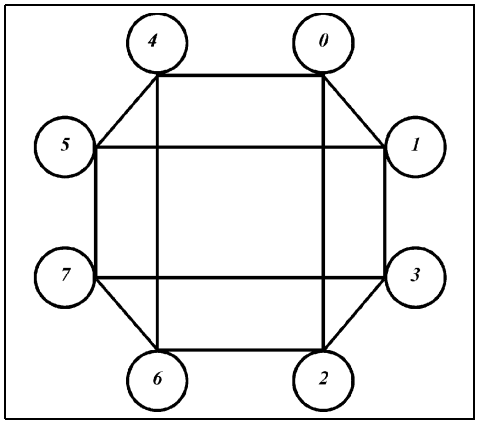


Рис. 1. Граф 3-куба

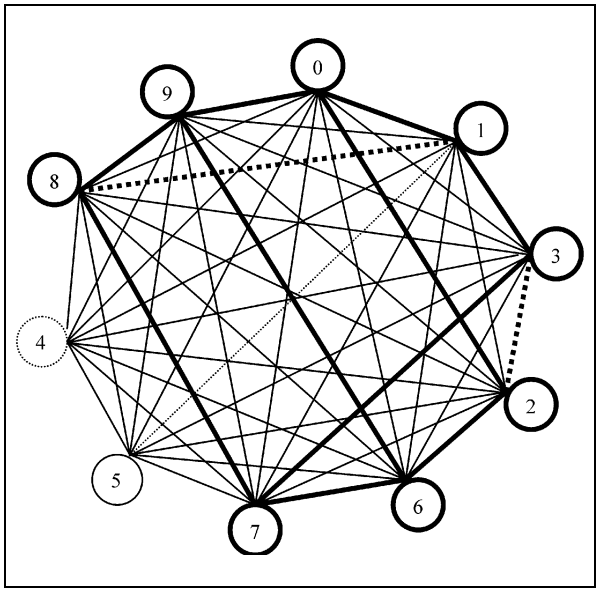


Рис. 2. Диагностический граф рабочей конфигурации № 8 2-ОУ системы

цикл между его восемью вершинами, а остальные дуги — внутренние. Двукратное применение алгоритмов синтеза [2] к графу G_0^8 позволяет построить 2-ОУ граф G_R^{10} (рис. 2) (смысл линий различной толщины, изображающих вершины и дуги, будет пояснен в Примере).

Каждая из конфигураций (структур) исправной m-ОУ системы, представленной графом G_R^{n+m}, включает в себя n рабочих и m резервных модулей (вершин графа). Для упрощения будем считать, что исходная РК содержит n смежных рабочих вершин, номера которых совпадают с номерами целевого графа, а добавленные (резервные) вершины также являются смежными. Последовательно сдвигая $(n+m)$ вершин на одну позицию, напри-

мер, против часовой стрелки, получим остальные $(n + m - 1)$ РК.

Для последующих действий представим информацию о графе G_R^{n+m} в виде таблицы, которую назовем первичной [5]. В ее левом столбце запишем номера $(n + m)$ РК, в следующих $(n + m)$ столбцах — составы этих РК, а затем — номера дуг, соответствующие номерам дуг исходной РК (т. е. номерам дуг целевого графа). Пример первичной таблицы, построенной для графа G_R^{10}, представлен в табл. 1. Она содержит восемь столбцов для номеров рабочих вершин, два столбца — для резервных вершин и пять столбцов — для номеров дуг целевого графа (подробнее о составе этих дуг — при обсуждении примера).

По мере функционирования m-ОУ системы и накопления в ней отказавших компонентов для их устранения из состава работающей системы целесообразно использовать РК с меньшим числом резервных вершин. Для систематического перебора таких РК нужна совокупность вторичных таблиц первого, второго, ..., m-го уровней, по структуре подобных первичной таблице, но содержащих меньше на одну, две, ..., m резервных вершин [5].

ПОСТАНОВКА ЗАДАЧИ

Пусть анализируемая m-ОУ система представлена диагностическим графом G_R^{n+m}, n вершин которого представляют n рабочих модулей (процессоров) системы, а m оставшихся вершин — резервные и (или) отказавшие модули. По построению, граф G_R^{n+m} — симметричный и однородный, но не обязательно полный.

Будем считать, что допускаются устойчивые кратные отказы ограниченного числа вершин и дуг. В отличие от работы [6] предполагается, что во время выполнения СД может возникнуть одиночный отказ. Предполагается, что неисправности модулей таковы, что не приводят к несанкционированному запуску СД в отказавших модулях и, следовательно, не искажают разработанную организацию СД.

Предполагается, что в исходной МВС есть диагностический монитор, который инициирует процессы контроля, самодиагностирования и осуществляет реконфигурацию по результатам СД состояния компонентов очередной РК.

Требуется разработать такой метод контроля и СД компонентов ОУ-системы, чтобы после выполнения СД каждый исправный модуль имел одинаковую диагностическую информацию о состоянии отказавших компонентов очередной РК системы, включая компоненты, отказавшие в процессе СД.

МЕТОД САМОДИАГНОСТИРОВАНИЯ ОТКАЗАВШИХ КОМПОНЕНТОВ

Модель БГМ и связанные с ней правила дешифрации результатов тестирования предполагают, что в процессе тестирования компонента его техническое состояние (исправное или неисправное) остается неизменным. Однако эта предпосылка может и не выполняться во время процесса СД. Но если отказ компонента возникнет в процессе СД, то одни МПщ будут тестировать его в исправном состоянии, а другие — в неисправном состоянии. Тогда дешифрация результатов тестирования может дать неправильный диагноз. Для обнаружения отказов, возникающих в процессе СД, и получения правильного диагноза будем дву- и более кратно запускать процесс СД и сравнивать результаты двух соседних запусков. Совпадение результатов первого и второго запусков свидетельствует о том, что неисправности компонентов, подозреваемых в отказе, имели место до первого запуска, а потому не привели к расхождению полу-

Таблица 1

Первичная таблица

№№ п/п	Номера рабочих вершин								Резервные вершины		Номера дуг				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	*0*	*1*	*3*	*2*	*6*	*7*	*5*	*4**	*8*	*9*	*4–0*	*0–2*	*3–7*	*6–4*	*5–1**
1	*1*	*3*	*2*	*6*	*7*	*5*	*4**	*8*	*9*	*0*	*8–1*	*1–6*	*2–5*	*7–8*	*4–3*
2	*3*	*2*	*6*	*7*	*5*	*4**	*8*	*9*	*0*	*1*	*9–3*	*3–7*	*6–4*	*5–9*	*8–2*
3	*2*	*6*	*7*	*5*	*4**	*8*	*9*	*0*	*1*	*3*	*0–2*	*2–5*	*7–8*	*4–0*	*9–6*
4	*6*	*7*	*5*	*4**	*8*	*9*	*0*	*1*	*3*	*2*	*1–6*	*6–4*	*5–9*	*8–1*	*0–7*
5	*7*	*5*	*4**	*8*	*9*	*0*	*1*	*3*	*2*	*6*	*3–7*	*7–8*	*4–0*	*9–3*	*1–5**
6	*5*	*4**	*8*	*9*	*0*	*1*	*3*	*2*	*6*	*7*	*2–5*	*5–9*	*8–1*	*0–2*	*3–4*
7	*4**	*8*	*9*	*0*	*1*	*3*	*2*	*6*	*7*	*5*	*6–4*	*4–0*	*9–3*	*1–6*	*2–8*
8	*8*	*9*	*0*	*1*	*3*	*2*	*6*	*7*	*5*	*4**	*7–8*	*8–1*	*0–2*	*3–7*	*6–9*
9	*9*	*0*	*1*	*3*	*2*	*6*	*7*	*5*	*4**	*8*	*5–9*	*9–3*	*1–6*	*2–5*	*7–0*

ченных результатов тестирования. Если же эти результаты не совпали, то в процессе первого или второго запусков возникли новые неисправности компонентов системы. В таком случае процесс СД нужно продолжить и повторять его до тех пор, пока не совпадут результаты проверок двух соседних запусков: второго и третьего, третьего и четвертого и т. д. Если учесть, что в высоконадежных системах неисправности компонентов бывают сравнительно редкими, то с большой вероятностью можно считать, что в течение двух—четырех запусков СД может быть не больше одного отказа. Эта особенность возникновения и проявления одиночного отказа и заложена в четырехкратном (максимум) повторении процесса СД. Отметим, что существует ненулевая (хотя и очень малая) вероятность того, что новый отказ возникнет в модуле в тот интервал времени, когда он исполняет функции проверяющего для своих преемников. В этом случае, скорее всего, процесс СД будет прерван, и спустя некоторое время диагностический монитор запустит процесс контроля вновь, но уже с другого начального модуля.

Контроль цифровой системы состоит в том, что каждый исправный модуль должен проверить состояние модулей, непосредственно с ним связанных (преемников), и соответствующих линий связи. Для оптимизации последовательности выполнения проверок одного модуля разными МПщ на диагностическом графе системы построим гамильтонов цикл. Для излагаемого ниже метода достаточно взять любой гамильтонов цикл, который обходит все вершины диагностического графа РК исправной системы. Такой цикл выбирается один раз при проектировании системы контроля.

Как и в работе [6], будем представлять информацию о составе выбранной РК № w системы в виде таблицы T_{Sw}. Она состоит из n строк T_{iw}, ..., T_j, ..., T_{nw}, в каждой из которых для модуля m_j ($j = i_w$, ..., n_w, где $0 \le i_w$, ..., $n_w \le n + m - 1$) записаны номера модулей-преемников m_{k1}, m_{k2}, ...; в ней также есть поля для записи результатов проверки модулем m_j этих преемников $r_{j,\,k1}$, $r_{j,\,k2}$, ... и значений их ранга при ранжировании графа. Пример такой таблицы T_{S0}, построенной для РК № 0 (см. табл. 1), показан в табл. 2. Как видно, в каждой из ее восьми строк заполнены только клетки для проверяющего модуля m_j и его преемников — m_{k1}, ..., m_{k3}.

Для отображения взаимного положения модулей в гамильтоновом цикле будем записывать в строке T_j преемников модуля m_j так, что модуль m_{k1} является соседом справа модуля m_j, который, в свою очередь, является соседом справа для своего последнего преемника (модуля m_{k3} в рассматриваемом примере). Для РК № 0 выбран такой гамильтонов цикл: $0 \to 1 \to 3 \to 2 \to 6 \to 7 \to 5 \to 4 \to 0$, именно в таком порядке в табл. 2 записаны строки T_j.

Традиционно процесс контроля запускается диагностическим монитором через определенные интервалы времени, что предотвращает накапливание отказавших компонентов в системе. Назначим начальной вершиной процесса контроля одну из вершин гамильтонова цикла в выбранной РК № w, обозначим ее m_n. Тогда по сигналу диагностического монитора МПщ m_n должен выполнить следующие действия:

- из таблицы T_{Sw} выделить строку T_n, найти в ней номера преемников m_{k1}, m_{k2}, ...; проверить их техническое состояние и в клетки $r_{n,\,k1}$, $r_{n,\,k2}$, ... строки T_n записать результаты проверок в двоичном коде: 0 — исправен, 1 — подозревается в отказе;
- оценить результаты проверок:
  — если результаты всех проверок — нулевые ($r_{n,\,k1} = 0$, $r_{n,\,k2} = 0$, ...), то передать модулю m_{k1} управление продолжением процесса контроля;

Таблица 2

Таблица состава T_{S0} для РК № 0

Символ строки	Проверяющий модуль и ранг		Преемники m_j, результаты их проверок и ранг								
	m_j	ранг m_j	m_{k1}	$r_{j,\,k1}$	ранг m_{k1}	m_{k2}	$r_{j,\,k2}$	ранг m_{k2}	m_{k3}	$r_{j,\,k3}$	ранг m_{k3}
T_0	0	–	1	–	–	2	–	–	4	–	–
T_1	1	–	3	–	–	5	–	–	0	–	–
T_3	3	–	2	–	–	7	–	–	1	–	–
T_2	2	–	6	–	–	0	–	–	3	–	–
T_6	6	–	7	–	–	4	–	–	2	–	–
T_7	7	–	5	–	–	3	–	–	6	–	–
T_5	5	–	4	–	–	1	–	–	7	–	–
T_4	4	–	0	–	–	6	–	–	5	–	–

— если среди полученных результатов проверок есть как единичные, так и нулевые (например, $r_{n,\,kj} = 0$), то передать одному из модулей с нулевым результатом (модулю m_{kj}) управление для перехода к процессу СД;
— если же проверки всех преемников m_n дали единичные результаты, то процесс контроля нужно начать с другого модуля (например, с модуля m_{n2}, находящегося в "середине" гамильтонова цикла).

Модуль m_{k1} и каждый из последующих модулей выполняют аналогичные действия. Этот процесс проверок и "хождения" по выбранному циклу заканчивается, когда управление возвращается к начальному модулю m_n (m_{n2}). Такой детерминированный путь обхода вершин графа возможен только при исправном состоянии всех компонентов проверяемой РК. Если же на одном из этапов контроля получен единичный результат, свидетельствующий об обнаружении одного или нескольких отказавших компонентов, то процесс контроля прерывается, и диагностический монитор переходит к процессу СД.

На основе информации, полученной в процессе контроля, монитор начинает первый запуск процесса СД с того модуля m_j, исправность которого подтверждена проверкой, выполненной тем же модулем m_i, который первым получил единичный результат при проверке другого преемника — модуля m_k. Начальный модуль m_j ($\neq m_n$, m_{n2}) должен выполнить следующие действия:

— из таблицы состава T_{Sw} выделить строку T_j, переписать ее в промежуточную таблицу PT_{j1} (в индексе символа указан номер j начального МПщ и номер запуска СД);
— в строке T_j найти номера преемников m_{k1}, m_{k2}, ...; проверить их состояние и в клетки $r_{j,\,k1}$, $r_{j,\,k2}$, ... строки T_j записать результаты проверок;
— по нулевым результатам проверок в строке T_j выделить подмножество номеров исправных преемников $\{q_j\}$ и первому из них передать таблицу PT_{j1} и функции МПщ на следующем шаге СД.

Если среди преемников m_j нет ни одного исправного, то процесс СД не может быть продолжен, и для получения диагностических результатов монитор должен запустить процесс контроля или СД с другого начального модуля.

Очередной исправный модуль m_u, получив управление процессом СД, должен выполнить следующие действия (*функции* МПщ):

1) из таблицы состава T_{Sw} выделить строку T_u, переписать ее в таблицу PT_{j1}, полученную от предшественника m_j (m_{u-1});

2) в строке T_u найти номера преемников m_{k1}, m_{k2}, ...; проверить их состояние и в клетки $r_{u,\,k1}$, $r_{u,\,k2}$, ... строки T_u записать результаты проверок;

3) по нулевым результатам проверок в строке T_u выделить подмножество исправных преемников $\{q_u\}$, сравнить номера $\{q_u\}$ с номерами модулей в левом столбце PT_{j1}:

— если один из номеров подмножества $\{q_u\}$ отсутствует в списке номеров левого столбца PT_{j1}, то передать соответствующему модулю (m_{u+1}) таблицу PT_{j1} и функции МПщ на следующем шаге; при этом модуль m_{u+1} вернется к п. 1 процесса СД;
— если два (или больше) номера из подмножества $\{q_u\}$ отсутствуют в списке номеров левого столбца PT_{j1}, то одному из модулей с этими номерами (например, m_{k1}) передать таблицу PT_{j1} и функции МПщ на следующем шаге;
— если все номера $\{q_u\}$ — в списке номеров левого столбца PT_{j1}, то перейти к п. 4;

4) найти кратчайший путь от модуля m_u к одному из исправных модулей m_{u+1}, m_{u+2}, ..., m_{u+v1}, еще не исполнявших функции МПщ, с помощью процедуры ранжирования (см. работы [6, 10]), передать найденному модулю m_{u+1} (m_{u+2}, ...) таблицу PT_{j1} и функции МПщ на следующем шаге; если при выполнении процедуры ранжирования выяснится, что все обнаруженные исправные модули исполняли функции МПщ, то это означает, что первый запуск СД завершен.

Если данный запуск СД был первым, то для подтверждения результатов проверок, накопленных в таблице PT_{j1}, и обнаружения возможных новых отказов диагностический монитор должен выполнить второй запуск СД с модуля m_{u+v1}, на котором закончился первый запуск.

Для совпадения описаний процедур, выполняемых в первом и втором запусках, модулю m_{u+v1} присвоим тот же символ m_j (начальный модуль второго запуска СД). Тогда модуль m_j должен выполнить такие действия:

— образовать новую таблицу PT_{j2} и переписать в нее из таблицы PT_{j1} строку T_{u+v1} с результатами проверок, полученными модулем m_{u+v1} в процессе первого запуска СД (в индексе символа данной таблицы указан номер j начального МПщ и номер запуска);
— по нулевым результатам проверок в строке T_j ($\equiv T_{u+v1}$) выделить подмножество номеров исправных преемников $\{q_j\}$ и первому из них (например, модулю m_u) передать таблицы PT_{j1}, PT_{j2} и функции МПщ на следующем

шаге СД. (В отличие от первого запуска такой исправный преемник всегда существует, так как модуль m_{u+v1} ($\equiv m_j$) получал управление в первом запуске процесса СД от исправного модуля.)

Очередной исправный МПщ m_u, получив управление процессом СД, должен выполнить те же действия, что и в первом запуске, за исключением того, что результаты всех проверок записываются в таблицу PT_{j2}, а не в таблицу PT_{j1}, содержимое которой сохраняется неизменным на протяжении второго запуска.

В процессе выполнения переходов по цепочке исправных модулей во втором запуске СД рано или поздно выяснится, что все обнаруженные исправные модули уже исполнили функции МПщ, т. е. процедуры второго запуска СД закончены. Тогда модуль m_{u+v2}, у которого во втором запуске совпали списки исправных и проверяющих модулей, должен сравнить содержимое таблиц PT_{j1} и PT_{j2}: сначала номера строк, затем результаты проверок $r_{j1,\,k}$, $r_{j2,\,k}$, ..., записанные в одноименных строках этих таблиц:

— в случае несовпадения — запустить процесс СД в третий (четвертый) раз, используя в качестве начального модуль m_{u+v2} (m_{u+v3}) и в качестве начальной таблицу PT_{j2} (PT_{j3});

— а при совпадении записей — передать в обратном направлении таблицу PT_{j2} (PT_{j3}, PT_{j4}) каждому из исправных модулей, перечисленных в ее левом столбце.

Для обработки информации из полученной таблицы PT_{j2} (PT_{j3}, PT_{j4}) каждый исправный модуль переписывает ее в диагностическую таблицу T_{Dw}, построенную заранее для РК № w. В строке D_k ($k = i_w, \ldots, n_w$) таблицы T_{Dw} для проверяемого модуля m_k записаны номера его предшественников (МПщ) m_{j1}, m_{j2}, m_{j3}; есть также клетки для записи результатов $r_{j1,\,k}$, $r_{j2,\,k}$, $r_{j3,\,k}$ его проверки этими модулями и клетки для записи диагностических заключений о состоянии модуля m_k и инцидентных ему линий связи, получаемых после обработки результатов $r_{j1,\,k}$, $r_{j2,\,k}$, $r_{j3,\,k}$. Для примера в табл. 3 показаны две такие таблицы, построенные для РК № 0 (см. табл. 1), содержащие по восемь строк. Они различаются записями в столбцах, озаглавленных “Результаты проверок” и “Диагностическое заключение”. Эти записи появляются в диагностической таблице после проведения СД.

Нетрудно заметить, что при $j = k$ преемники модуля m_j, показанные в строке T_j табл. 2, совпадают с предшественниками модуля m_k, приведенными в левой части строки D_k табл. 3, *а* и 3, *б*. Это

Таблица 3

Диагностические таблицы для РК № 0

Символ строки	Предшественники m_k			Проверяемый модуль m_k	Результаты проверок			Диагностическое заключение			
	m_{j1}	m_{j2}	m_{j3}		$r_{j1,k}$	$r_{j2,k}$	$r_{j3,k}$	s_k	$a_{j1,k}$	$a_{j2,k}$	$a_{j3,k}$
а. Диагностическая таблица T_{D0} для таблицы PT_{53}											
D_0	*1*	*2*	*4*	*0*	0	0	–	0	0	0	–
D_1	*3*	*5*	*0*	*1*	0	1	0	0	0	1	0
D_3	*2*	*7*	*1*	*3*	0	0	0	0	0	0	0
D_2	*6*	*0*	*3*	*2*	0	0	0	0	0	0	0
D_6	*7*	*4*	*2*	*6*	0	–	0	0	0	–	0
D_7	*5*	*3*	*6*	*7*	0	0	0	0	0	0	0
D_5	*4*	*1*	*7*	*5*	–	1	0	0	–	1	0
D_4	*0*	*6*	*5*	*4*	1	1	1	1	*z*	*z*	*z*
б. Диагностическая таблица T^1_{D0} для таблицы PT_{31}											
D_0	*1*	*2*	*4*	*0*	0	0	–	0	0	0	–
D_1	*3*	*5*	*0*	*1*	0	1	0	0	0	1	0
D_3	*2*	*7*	*1*	*3*	0	0	0	0	0	0	0
D_2	*6*	*0*	*3*	*2*	0	0	0	0	0	0	0
D_6	*7*	*4*	*2*	*6*	0	–	0	0	0	–	0
D_7	*5*	*3*	*6*	*7*	0	0	0	0	0	0	0
D_5	*4*	*1*	*7*	*5*	–	1	0	0	–	1	0
D_4	*0*	*6*	*5*	*4*	1	0	1	0	1	0	1

совпадение — не случайно, оно следует из того, что в системе используются двунаправленные связи, поэтому модули m_{k1}, m_{k2}, m_{k3} — преемники модуля m_j — являются также его предшественниками m_{j1}, m_{j2}, m_{j3}.

Процедура обработки исправным модулем информации из таблицы PT_{j2} (PT_{j3}, PT_{j4}) заключается в следующем:

1) из строк T_j, ..., T_u, ..., T_{u+vi} ($i = 2, 3, 4$) таблицы PT_{j2} (PT_{j3}, PT_{j4}) выделить результаты проверок $r_{j,k1}$, ..., $r_{u,k1}$, ..., $r_{u+vi,k1}$ и записать их в соответствующие клетки строк T_D таблицы T_{Dw}; в оставшихся клетках этих столбцов сделать прочерки (—), так как результатов этих проверок нет;

2) проанализировать результаты проверок, содержащиеся в строке D_k таблицы T_{Dw}, и записать в клетки s_k, $a_{j1,k}$, $a_{j2,k}$, ... этой строки символы диагностического заключения, полученные по следующим правилам:

- если в клетках результатов проверок строки D_k есть хотя бы один нуль (например, $r_{j1,k} = 0$), то записать 0 в клетки на пересечениях строки D_k со столбцами s_k и $a_{j1,k}$;
- если в этой же строке D_k есть еще 0 (1), то записать 0 (1) в клетки на пересечениях строки D_k со столбцами переменных соответствующих линий;
- если в клетках результатов проверок строки D_k записаны только единицы, то записать 1 в клетку на пересечении строки D_k со столбцом s_k и записать буквы z в клетки на пересечениях строки D_k со столбцами переменных соответствующих линий;
- в оставшихся клетках этих столбцов сделать прочерки (—).

Теперь расшифровка информации из таблицы PT_{j2} (PT_{j3}, PT_{j4}) сводится к замене символов 0, 1, (—) и z, записанных в клетках строки D_k таблицы T_{Dw}, на заключения о состояниях модуля m_k и инцидентных линий связи согласно правилам: 0 → "модуль исправен", "линия исправна"; 1 → "модуль (линия) подозревается в отказе"; (—) неопределенное состояние линии связи, потому что соответствующая проверка не выполнялась; z → неопределенное состояние линии связи, так как отказавший модуль "маскирует" состояние инцидентной линии связи.

Поскольку каждый исправный модуль получил одинаковую информацию в виде таблицы PT_{j2} (PT_{j3}, PT_{j4}), то после ее обработки все эти модули будут иметь одинаковые диагностические заключения о состоянии компонентов системы.

ПРОЦЕДУРА РЕКОНФИГУРАЦИИ

Обработка результатов проверок из таблицы PT_{j2} (PT_{j3}, PT_{j4}) определяет подмножество компонентов в составе РК № w, подозреваемых в отказе. Процедура реконфигурации m-ОУ системы сводится к нахождению в первичной таблице (или в одной из вторичных таблиц) такой РК № w^*, в состав которой не входят компоненты, подозреваемые в отказе: т. е. отказавшая вершина займет место одной из резервных, отказавшая дуга не будет соединять пару смежных рабочих вершин и не будет входить в состав дуг, перечисленных в правой части строки w^*.

После нахождения такой работоспособной РК № w^* диагностический монитор модифицирует таблицы T_{Sw} и T_{Dw}, заменив в них номера рабочих модулей прежней РК № w на соответствующие номера новой РК № w^*. Затем он запускает процесс контроля РК № w^*. Если контроль РК № w^* подтвердит ее работоспособность, то m-ОУ система возвращается к выполнению своих функциональных задач. В противном случае монитор вновь запустит процесс СД для идентификации тех компонентов, отказы которых нарушили работоспособность РК № w^*.

ПРИМЕР

Особенности процессов СД и устранения возникающих отказов покажем на примере той же 2-ОУ системы, построенной для целевой системы со структурой трехмерного куба, что и в работе [6]. Рассматривается 3-куб, в вершинах которого расположены процессорные элементы. На рис. 1 они обозначены кружками, в которых указаны их номера. Двукратное применение алгоритмов синтеза [2] к графу 3-куба G_0^8 (см. рис. 1) позволяет построить 2-ОУ граф G_R^{10} (см. рис. 2); подробности применения алгоритмов синтеза к графу G_0^8 можно найти в статье [5]. Первичная таблица РК графа G_R^{10} представлена табл. 1. Она состоит из 10-ти строк, соответствующих десяти РК и обозначенных номерами 0, 1; ..., 9; клетки столбцов 1, 2, ..., 8 каждой строки содержат номера рабочих вершин данной РК, клетки столбцов 9 и 10 — номера резервных вершин. В клетках столбцов 12—15 приведены номера дуг, соответствующие номерам четырех внутренних дуг целевого графа G_0^8. Начальная РК № 0 показана в строке 0: рабочие вершины *0*, *1*, ..., *5*, *4*, резервные *8*, *9*, рабочие дуги (*0—2*), (*3—7*), (*6—4*), (*5—1*). В столбце 11 показана дуга (*4—0*), которая замыкает гамильтонов цикл из рабочих вершин: $0 \to 1 \to 3 \to 2 \to 6 \to 7 \to 5 \to 4 \to 0$.

Дуги между смежными вершинами этого цикла в табл. 1 отдельно не показаны, так как их номера однозначно определяются номерами соседних столбцов: (*0—1*), (*1—3*) и т. д. Номера вершин РК, показанные в строках 1—9, получены циклическим сдвигом справа налево номеров вершин предыдущей строки *0*, *1*, ..., *8*. Номера дуг, показанные в клетках столбцов *11—15* каждой строки, получены путем замены номеров рабочих вершин строки 0 номерами вершин, записанными в тех же столбцах очередной РК. Поэтому в столбце 12 дуга (*0—2*) из строки 0 последовательно заменяется дугой, соединяющей вершины с номерами (*1—6*), (*3—7*) и т. д., стоящими в соответствующих столбцах ниже вершин *0* и *2*. Таким же образом получены номера других рабочих дуг.

Отметим, что алгоритм *A*2 оптимального обхода и тестирования вершинных процессорных элементов [11] разработан для проверки работоспособности вершин и дуг неизбыточного *n*-куба. Поэтому его применение для проверки 2-ОУ системы, построенной на основе 3-куба, возможно только для отдельных РК, содержащих 8 вершин и 12 дуг. Нетрудно видеть, что 2-ОУ граф G_R^{10} (см. рис. 2) содержит большое число таких РК. Поэтому вместо перебора всевозможных РК применим описанный процесс последовательного контроля компонентов РК № 0 и тех РК, в которые переходит система для исключения отказавших компонентов.

Итак, пусть 2-ОУ система работает в начальной РК № 0 (см. табл. 1). Предположим, что между циклами контроля отказала линия связи (*5—1*) (отмечена знаком * в клетках тех РК табл. 1, в состав которых она входит). Контроль работоспособности 2-ОУ системы начинается с модуля *0*, соответствующего начальной вершине выбранного гамильтонова цикла для РК № 0.

По сигналу диагностического монитора начальный модуль m_0 выполнит такие действия: из таблицы T_{S0} состава РК № 0 (см. табл. 2) выделит строку T_0, найдет в ней номера преемников m_{k1} (≡1), m_{k2} (≡2), m_{k3} (≡4), проверит их состояние и в клетки $r_{j,\,k1}$, $r_{j,\,k2}$, $r_{j,\,k3}$ строки T_0 запишет результаты проверок: $r_{0,\,1} = 0$, $r_{0,\,2} = 0$, $r_{0,\,4} = 0$ (см. табл. 4); так как результаты всех проверок — нулевые, то передаст модулю *1* (≡m_{k1}) управление для продолжения контроля (здесь и в последующем полужирным шрифтом выделен тот нуль, по которому передается управление следующему МПщ для продолжения контроля или СД).

Модуль *1*, получив управление от модуля *0*:

— из таблицы T_{S0} выделит строку T_1, найдет в ней номера преемников *3*, *5*, *0*, проверит их и в соответствующие клетки строки T_1 запишет результаты проверок: $r_{1,\,3} = 0$, $r_{1,\,5} = 1$, $r_{1,\,0} = 0$ (для сокращения числа таблиц будем записывать результаты проверок, находящиеся в разных модулях, в общую табл. 4);

— оценит результаты проверок и, обнаружив $r_{1,\,5} = 1$ и $r_{1,\,3} = 0$, передаст модулю *3* управление для перехода к процессу СД.

Начальный модуль *3* первого запуска процесса СД выполнит такие действия:

— из таблицы состава T_{S0} выделит строку T_3, перепишет ее в новую таблицу PT_{31} (в индексе символа таблицы указан номер *3* начального МПщ для первого запуска СД);

— в строке T_3 найдет номера преемников *2*, *7*, *1*; проверит их состояние и в соответствующие клетки строки T_3 запишет результаты проверок;

— по нулевым результатам проверок в строке T_3 выделит номера исправных преемников {$q_3 = 2, 7, 1$} и первому из них, модулю *2*, передаст таблицу PT_{31} и функции МПщ на следующем шаге СД.

В свою очередь, модуль *2* выполнит такие действия:

1) из таблицы T_{S0} выделит строку T_2 и перепишет ее в таблицу PT_{31} (см. табл. 4);

2) в строке T_2 найдет номера преемников *6*, *0*, *3*; проверит их и в соответствующие клетки строки T_2 запишет результаты проверок;

3) по нулевым результатам проверок в строке T_2 выделит номера исправных преемников {$q_2 = 6, 0, 3$}, сравнит их с номерами *3*, *2* в левом столбце таблицы PT_{31} и обнаружит, что модули *6* и *0* не исполняли функций МПщ, поэтому передаст первому из них (модулю *6*) таблицу PT_{31} с результатами проверок и функции МПщ на следующем шаге СД, при этом модуль *6* вернется к п. 1 процесса СД.

Аналогичные действия в роли МПщ поочередно выполнят модули *6*, *7*, результаты их проверок своих преемников показаны в строках T_6, T_7, табл. 4. Предположим, что после того, как модуль *6* проверил своих преемников и передал таблицу PT_{31} и управление модулю *7*, *модуль 4 отказал*. Получив управление от модуля *7*, модуль *5*, выполняя функции МПщ, проверит своих преемников и обнаружит, что его исправный преемник — модуль *7* уже исполнял функции МПщ. Поэтому модуль *5* перейдет к поиску пути к одному из исправных модулей, еще не исполнявших функции МПщ, с помощью процедуры ранжирования.

Из результатов проверок, записанных в строках T_3, ..., T_5 таблицы PT_{31}, модуль *5* определит, что функции МПщ не исполняли исправные модули *1* и *0*, из таблицы T_{S0} (см. табл. 2) выделит строки T_1

и T_0, добавит их в таблицу PT_{31}. Затем, выполняя процедуру ранжирования, модуль *5* присвоит ранг 0 модулю *5*, ранг 1 модулю *7*, затем ранг 2 модулям *3* и *6*, ранг 3 модулям *1* и *2*. На этом ранжирование будет закончено, так как модуль *1* попал в число ранжированных. (Результаты ранжирования показаны в строках таблицы PT_{31}; в некоторых клетках столбцов, "ранг m_{k1}" и "ранг m_{k2}" стоит знак *, означающий, что соответствующие модули m_{k1} и m_{k2} не участвуют в формировании искомого пути, так как они признаны отказавшими из-за единичных результатов их проверок.) Теперь модуль *5* передаст таблицу PT_{31} и функции МПщ на следующем шаге СД модулю *1* через цепочку модулей: $5 \rightarrow 7 \rightarrow 3 \rightarrow 1$.

Модуль *1*, получив управление от модуля *5*:

1) в строке T_1 таблицы PT_{31} (см. табл. 4) найдет номера преемников *3*, *5*, *0*; проверит их и в соответствующие клетки строки T_1 запишет результаты проверок;

2) по нулевым результатам проверок в строке T_1 выделит номера исправных преемников $\{q_1 = 3, 0\}$ и передаст таблицу PT_{31} и функции МПщ на следующем шаге СД модулю *0*, который после исполнения функций МПщ обнаружит, что исправных модулей, не исполнявших функции МПщ, нет, поэтому первый запуск СД завершен.

Для подтверждения результатов проверок, накопленных в таблице PT_{31}, и обнаружения возможных новых отказов монитор должен запустить процесс СД второй раз с модуля *0*, на котором за-

Таблица 4

Информация для РК № 0

Символ строки	Проверяющий модуль		Проверяемые модули, результаты их проверок и ранг								
	m_j	ранг m_j	m_{k1}	$r_{j,k1}$	ранг m_{k1}	m_{k2}	$r_{j,k2}$	ранг m_{k2}	m_{k3}	$r_{j,k3}$	ранг m_{k3}
Строки с результатами контроля РК № 0											
T_0	*0*	–	*1*	**0**	–	*2*	0	–	*4*	0	–
T_1	*1*	–	*3*	**0**	–	*5*	1	–	*0*	0	–
Таблица PT_{31}											
T_3	*3*	2	*2*	**0**	3	*7*	0	1	*1*	0	3
T_2	*2*	3	*6*	**0**	4	*0*	0	4	*3*	0	2
T_6	*6*	2	*7*	**0**	1	*4*	0	3	*2*	0	3
T_7	*7*	1	*5*	**0**	0	*3*	0	2	*6*	0	2
T_5	*5*	0	*4*	1	*	*1*	1	*	*7*	0	1
T_1	*1*	3	*3*	0	–	*5*	1	–	*0*	**0**	–
T_0	*0*	–	*1*	0	–	*2*	0	–	*4*	1	–
Таблица PT_{02}											
T_0	*0*	–	*1*	**0**	–	*2*	0	–	*4*	1	–
T_1	*1*	–	*3*	**0**	–	*5*	1	–	*0*	0	–
T_3	*3*	–	*2*	**0**	–	*7*	0	–	*1*	0	–
T_2	*2*	–	*6*	**0**	–	*0*	0	–	*3*	0	–
T_6	*6*	–	*7*	**0**	–	*4*	1	–	*2*	0	–
T_7	*7*	–	*5*	**0**	–	*3*	0	–	*6*	0	–
T_5	*5*	–	*4*	1	–	*1*	1	–	*7*	0	–
Таблица PT_{53}											
T_5	*5*	–	*4*	1	–	*1*	1	–	*7*	**0**	–
T_7	*7*	–	*5*	0	–	*3*	0	–	*6*	**0**	–
T_0	*6*	–	*7*	0	–	*4*	1	–	*2*	**0**	–
T_2	*2*	–	*6*	0	–	*0*	0	–	*3*	**0**	–
T_3	*3*	–	*2*	0	–	*7*	0	–	*1*	**0**	–
T_1	*1*	–	*3*	0	–	*5*	1	–	*0*	**0**	–
T_0	*0*	–	*1*	0	–	*2*	0	–	*4*	1	–

кончился первый запуск. В начале второго запуска модуль *0* выполнит такие действия:

— образует новую таблицу PT_{02} (см. табл. 4) и перепишет в нее из таблицы PT_{31} строку T_0 с результатами проверок, полученными в первом запуске СД (в индексе символа таблицы PT_{02} указан номер *0* начального МПщ во втором запуске);

— по нулевым результатам проверок в строке T_0 выделит номера исправных преемников $\{q_0 = 1, 2\}$ и передаст первому из них (модулю *1*) таблицы PT_{31}, PT_{02} и функции МПщ на следующем шаге СД.

Исправный модуль *1*, а затем модули *3*, *2*, *6*, *7*, *5*, получив управление процессом СД, выполняют те же действия, что и в первом запуске, но результаты всех проверок записывают в таблицу PT_{02} (см. табл. 4), а содержимое PT_{31} передают неизменным на протяжении второго запуска. Модуль *5* обнаружит, что исправных модулей, не исполнявших функции МПщ, больше нет, поэтому второй запуск СД завершен.

Сравнив записи в таблицах PT_{31} и PT_{02}, модуль *5* определит их различие по результатам проверки модуля *4*: в таблице PT_{31} $r_{6,4} = 0$, а в таблице PT_{02} $r_{6,4} = 1$; поэтому монитор запустит процесс СД в третий раз, используя в качестве начального модуль *5* и в качестве начальной таблицу PT_{02}.

Результаты проверок, полученных в третьем запуске СД с модуля *5*, показаны в строках T_5, ..., T_0 таблицы PT_{53} (в нижней части табл. 4). Особенность третьего запуска: не потребовалась процедура ранжирования, так как управление СД каждый раз передавалось последнему из исправных преемников очередного МПщ. После завершения третьего запуска модуль *0*, сравнив таблицы PT_{02} и PT_{53}, обнаружит их полное совпадение как по номерам строк, так и по результатам проверок в этих строках. Поэтому модуль *0* передаст таблицу PT_{53} каждому из исправных модулей по цепочке: $0 \to 1 \to 3 \to 2 \to 6 \to 7 \to 5$.

Теперь каждый исправный модуль РК № 0 системы выполнит следующую обработку информации из таблицы PT_{53}: из строк T_5, ..., T_0 выделит результаты проверок и запишет их в клетки строк T_{D0} (см. табл. 3, *а*); в соответствии с правилами, приведенными в основном тексте статьи, проанализирует результаты проверок, содержащиеся в каждой строке D_k таблицы T_{D0}, и запишет в клетки s_k, $a_{j1,k}$, $a_{j2,k}$, ... этой же строки символы соответствующего диагностического заключения.

Расшифровка символов, показанных в клетках столбцов "Диагностическое заключение" таблицы T_{D0}, позволяет прийти к такому диагнозу:

— подозреваются в отказе модуль *4* и линии связи (*5—1*) и (*1—5*);

— нет информации о состоянии трех линий, отмеченных прочерками (—): отказавший модуль *4* не использовался в качестве МПщ;

— нет информации о состоянии трех линий, отмеченных символом z в строке D_4, так как отказавший модуль *4* "маскирует" отказы своих входных линий;

— в РК № 0 исправны 7 модулей (*0*, *1*, *2*, *3*, *5*, *6*, *7*) и 16 линий связи, соответствующих тем клеткам в столбцах $a_{j1,k}$, ..., $a_{j3,k}$, в которых стоят нули.

Сравнение полученного диагноза с составом заявленной отказовой ситуации показывает, что вместо линии (*5—1*) подозревается в отказе пара линий: (*5—1*) и (*1—5*); причина в том, что логическими методами нельзя различить отказы линий, участвующих в инверсных проверках.

В табл. 3, *б* показана диагностическая таблица T^1_{D0}, построенная для информации из таблицы PT_{31}. Сравнение диагностических заключений, приведенных в табл. 3, *а* и 3, *б*, показывает различие записей в строке D_4: если в таблице T_{D0} модуль *4* подозревается в отказе, то в таблице T^1_{D0} модуль *4* признан исправным, а подозреваются в отказе линии (*0—4*) и (*5—4*). Причина такого диагноза в том, что по результату проверки модуля *4*, выполненной модулем *6* до его отказа, он был признан (естественно) исправным, а единичные результаты его последующих проверок модулями *5* и *0* (после отказа) были отнесены к отказам линий (*5—4*) и (*0—4*). Таким образом, если бы процесс СД был ограничен одним запуском, то получили бы неправильный диагноз.

Для устранения подозреваемых компонентов: модуля *4* и дуг (*5—1*), (*1—5*) диагностический монитор должен перевести 2-ОУ систему из РК № 0 в такую РК, в которую не входят эти компоненты. Обратившись к первичной таблице (см. табл. 1), нетрудно обнаружить, что модуль *4* входит в состав восьми РК — от РК № 0 до РК № 7 — и не входит только в РК № № 8 и 9, где он записан в числе резервных модулей; в эти же РК не входит и дуга (*5—1*). Для определенности, в качестве следующей выберем РК № 8. На рис. 2 представлен граф РК № 8: рабочие вершины и дуги показаны толстыми линиями, резервная вершина *5* и резервные дуги — тонкими линиями, а неисправная вершина *4* и пара дуг (*5—1*) и (*1—5*) — тонкими пунктирными линиями.

После смены РК для РК № 8 получим такой гамильтонов цикл: $8 \to 9 \to 0 \to 1 \to 3 \to 2 \to 6 \to \to 7 \to 8$. С учетом этого цикла монитор модифицирует таблицы T_{S0} и T_{D0}, заменит номера рабочих модулей в соответствии с подстановкой (*89013267*)

вместо (*01326754*) и получит таблицу T_{S8} (табл. 5) и диагностическую таблицу T_{D8} (табл. 6). Затем монитор запустит процесс контроля РК № 8, который подтвердит ее работоспособность.

Несколько слов об устранении компонентов, показанных в диагностической таблице T^1_{D0} таблицы для PT_{31} (см. табл. 3, *б*). Как видно из первичной табл. 1, есть только три РК: №№ 7, 8 и 9, в состав которых не входят подозреваемые линии (*5—1*), (*1—5*), (*0—4*) и (*5—4*). Если монитор выберет РК № 8, то функционирование системы совпадет с описанным. Если монитор выберет РК № 7, то контроль признает ее неработоспособной, так как среди рабочих модулей есть отказавший модуль *4*, и запустит процесс СД. На основании нового диагноза, который будет правильным, монитор переведет систему в РК № 8.

Несколько иная ситуация будет при переходе в РК № 9: действительно отказавший модуль *4* попадет на место резервной вершины (в столбце 9), и последующий контроль признает РК № 9 работоспособной. Однако, поставив на место резервного отказавший модуль *4*, мы фактически оставили систему без резерва. Поэтому система продолжит рабочее функционирование до первого отказа любого компонента РК № 9. Таким образом, учет неправильного диагноза приведет к преждевременному и неэффективному расходованию резервных компонентов и снижению времени надежного функционирования системы.

Если бы спустя некоторое время работы в РК № 8 возникли отказы модуля *3* и линии (*6—9*), то последующие действия системы по организации контроля, СД и реконфигурации можно найти в работе [6]. Здесь же будем считать, что между соседними включениями системы контроля произошел отказ линии (*8—1*), а затем в процессе первого запуска СД между проверками модуля *1* модулем *3* и модуля *3* модулем *2* произошел отказ линии (*2—3*) (эти линии показаны на рис. 2 толстым пунктиром).

Итак, по сигналу диагностического монитора модуль *8* начнет контроль работоспособности РК № 8: выполнит действия, аналогичные действиям модуля *0* в РК № 0, но искомую строку T_8 возьмет

Таблица 5

Таблица состава T_{S8} для РК № 8

Символ строки	Проверяющий модуль и ранг		Преемники m_j, результаты их проверок и ранг								
	m_j	ранг m_j	m_{k1}	$r_{j,\,k1}$	ранг m_{k1}	m_{k2}	$r_{j,\,k2}$	ранг m_{k2}	m_{k3}	$r_{j,\,k3}$	ранг m_{k3}
T_8	*8*	–	*9*	–	–	*1*	–	–	7	–	–
T_9	*9*	–	*0*	–	–	*6*	–	–	8	–	–
T_0	*0*	–	*1*	–	–	*2*	–	–	9	–	–
T_1	*1*	–	*3*	–	–	*8*	–	–	0	–	–
T_3	*3*	–	*2*	–	–	*7*	–	–	1	–	–
T_2	*2*	–	*6*	–	–	*0*	–	–	3	–	–
T_6	*6*	–	*7*	–	–	*9*	–	–	2	–	–
T_7	*7*	–	*8*	–	–	*3*	–	–	6	–	–

Таблица 6

Диагностическая таблица T_{D8} для таблицы PT_{23}

Символ строки	Предшественники m_k			Проверяемый модуль m_k	Результаты проверок			Диагностическое заключение			
	m_{j1}	m_{j2}	m_{j3}		$r_{j1,k}$	$r_{j2,k}$	$r_{j3,k}$	s_k	$a_{j1,k}$	$a_{j2,k}$	$a_{j3,k}$
D_8	*9*	*1*	*7*	*8*	0	1	0	0	0	1	0
D_9	*0*	*6*	*8*	*9*	0	0	0	0	0	0	0
D_0	*1*	*2*	*9*	*0*	0	0	0	0	0	0	0
D_1	*3*	*8*	*0*	*1*	0	1	0	0	0	1	0
D_3	*2*	*7*	*1*	*3*	1	0	0	0	1	0	0
D_2	*6*	*0*	*3*	*2*	0	0	1	0	0	0	1
D_6	*7*	*9*	*2*	*6*	0	0	0	0	0	0	0
D_7	*8*	*3*	*6*	*7*	0	0	0	0	0	0	0

из таблицы TS_8 (см. табл. 5) (результаты проверок, выполненных модулем *8*, показаны в строке T_8 табл. 7); обнаружив $r_{8,1} = 1$ и $r_{8,9} = 0$, модуль *8* переведет монитор в режим СД.

Начальный модуль *9* первого запуска СД: из таблицы T_{S8} (см. табл. 5) выделит строку T_9, перепишет ее в промежуточную таблицу PT_{91}; в строке T_9 найдет номера преемников *0*, *6*, *8*; проверит их состояние и в соответствующие клетки строки T_9 запишет результаты проверок; затем по нулевым результатам проверок выделит номера исправных преемников $\{q_9 = 0, 6, 8\}$ и первому из них, модулю *0*, передаст таблицу PT_{91} и функции МПщ на следующем шаге СД (результаты проверок, выполненных модулем *9*, показаны в строке T_9 таблицы PT_{91} (см. табл. 7)).

Действия модулей *0*, *1*, ..., *8* подобны действиям МПщ, ранее описанным в процессе СД в РК № 0, за исключением того, что необходимые строки будут извлекаться из табл. T_{S8} (см. табл. 5); а результаты выполненных ими проверок записываться в строки таблицы PT_{91} (см. табл. 7). Отметим, что после того, как модуль *3* передал таблицу PT_{91} и функции МПщ модулю *2*, *отказала линия* (*2—3*), что нашло отражение в результате проверки $r_{2,3} = 1$. В конце запуска модуль *8*, выполняя функции

Таблица 7

Информация для РК № 8

Символ строки	Проверяющий модуль и ранг		Преемники m_j, результаты их проверок и ранг								
	m_j	ранг m_j	m_{k1}	$r_{j,k1}$	ранг m_{k1}	m_{k2}	$r_{j,k2}$	ранг m_{k2}	m_{k3}	$r_{j,k3}$	ранг m_{k3}
Строки с результатами контроля РК № 8											
T_8	*8*	–	*9*	**0**	–	*1*	1	–	*7*	0	–
Таблица PT_{91}											
T_9	*9*	–	*0*	**0**	–	*6*	0	–	*8*	0	–
T_0	*0*	–	*1*	**0**	–	*2*	0	–	*9*	0	–
T_1	*1*	–	*3*	**0**	–	*8*	1	–	*0*	0	–
T_3	*3*	–	*2*	**0**	–	*7*	0	–	*1*	0	–
T_2	*2*	–	*6*	**0**	–	*0*	0	–	*3*	1	–
T_6	*6*	–	*7*	**0**	–	*9*	0	–	*2*	0	–
T_7	*7*	–	*8*	**0**	–	*3*	0	–	*6*	0	–
T_8	*8*	–	*9*	**0**	–	*1*	1	–	*7*	0	–
Таблица PT_{82}											
T_8	*8*	–	*9*	**0**	–	*1*	1	–	*7*	0	–
T_9	*9*	–	*0*	**0**	–	*6*	0	–	*8*	0	–
T_0	*0*	–	*1*	**0**	–	*2*	0	–	*9*	0	–
T_1	*1*	–	*3*	**0**	–	*8*	1	–	*0*	0	–
T_3	*3*	–	*2*	1	–	*7*	**0**	–	*1*	0	–
T_7	*7*	–	*8*	0	–	*3*	0	–	*6*	**0**	–
T_6	*6*	–	*7*	0	–	*9*	0	–	*2*	**0**	–
T_2	*2*	–	*6*	0	–	*0*	0	–	*3*	1	–
Таблица PT_{23}											
T_2	*2*	–	*6*	**0**	–	*0*	0	–	*3*	1	–
T_6	*6*	–	*7*	**0**	–	*9*	0	–	*2*	0	–
T_7	*7*	–	*8*	**0**	–	*3*	0	–	*6*	0	–
T_8	*8*	–	*9*	**0**	–	*1*	1	–	*7*	0	–
T_9	*9*	–	*0*	**0**	–	*6*	0	–	*8*	0	–
T_0	*0*	–	*1*	**0**	–	*2*	0	–	*9*	0	–
T_1	*1*	–	*3*	**0**	–	*8*	1	–	*0*	0	–
T_3	*3*	–	*2*	1	–	*7*	0	–	*1*	0	–

МПщ, проверит своих преемников и обнаружит, что его исправные преемники (модули *9* и *7*) уже исполняли функции МПщ, перейдет к процедуре ранжирования, в начале которой обнаружит, что больше нет исправных модулей для исполнения функций МПщ; таким образом, первый запуск СД завершен.

В начале второго запуска СД модуль *8*, на котором закончился первый запуск, образует таблицу PT_{82} (см. табл. 7), перепишет в нее из таблицы PT_{91} строку T_8 с результатами проверок, полученными в первом запуске СД, по нулевым результатам проверок в строке T_8 выделит номера исправных преемников $\{q_8 = 9, 7\}$ и передаст первому из них, модулю *9*, таблицы PT_{91}, PT_{82} и функции МПщ на следующем шаге СД.

Исправный модуль *9*, а затем модули *0*, *1*, ..., *6*, *2*, получив управление процессом СД, выполняют те же действия, что и в первом запуске, но результаты всех проверок записывают в таблицу PT_{82} (см. табл. 7), а содержимое таблицы PT_{91} передают неизменным на протяжении второго запуска; номер модуля, которому передаются функции МПщ на следующем шаге, отмечен в таблицу PT_{82} полужирным нулем в соседней клетке результата проверки. Модуль *2* обнаружит, что исправных модулей, не исполнявших функции МПщ, больше нет, поэтому второй запуск СД завершен.

Сравнив записи в таблицах PT_{91} и PT_{82}, модуль *2* определит их различие по результатам проверки $r_{3,2}$: в таблице PT_{91} $r_{3,2} = 0$, а в таблице PT_{82} $r_{3,2} = 1$; поэтому диагностический монитор запустит процесс СД в третий раз, используя в качестве начального модуль *2* и в качестве начальной таблицу PT_{82}.

Результаты проверок, полученных в третьем запуске СД с модуля *2*, показаны в строках T_2, ..., T_3 таблицы PT_{23} (в нижней части табл. 7). После завершения третьего запуска модуль *3*, сравнив таблицы PT_{82} и PT_{23}, обнаружит их полное совпадение и передаст таблицу PT_{23} каждому из исправных модулей по цепочке $3 \to 1 \to 0 \to 9 \to 8 \to 7 \to 6 \to 2$.

Затем каждый исправный модуль РК № 8, получив таблицу PT_{23}, выделит из ее строк результаты проверок и запишет их в клетки строк таблицы T_{D8} (см. табл. 6); в соответствии с правилами, приведенными в основном тексте статьи, проанализирует их и запишет в клетки s_k, $a_{j1,k}$, $a_{j2,k}$,... каждой строки D_k соответствующие символы. Их расшифровка приводит к следующим диагностическим заключениям:

— подозреваются в отказе пары линий связи (*1—8*) и (*8—1*),(*2—3*) и (*3—2*);
— в РК № 8 исправны все восемь модулей и остальные линии связи, кроме перечисленных.

Сравнение полученного диагноза с составом исходной отказовой ситуации показывает, что вместо линий (*2—3*) и (*8—1*) подозреваются в отказе указанные пары линий; причина та же, что и при диагностировании линии (*5—1*) в РК № 0.

Для исключения подозреваемых компонентов из состава РК диагностический монитор должен перевести 2-ОУ систему в такую РК, в которую не входят эти компоненты, а также ранее отказавшие модуль *4* и линии (*5—1*), (*1—5*). Для поиска такой РК целесообразно воспользоваться вторичной таблицей, построенной на основании РК № 8. Такая таблица представлена в табл. 8 и отличается от первичной таблицы (см. табл. 1) тем, что имеет только один столбец (№ 9) для резервной вершины, а столбец 10 закреплен за отказавшим модулем *4*. Отметим в клетках табл. 9 подозреваемые компоненты: линию (*8—1*) знаком *, а другую линию (*2—3*), связывающую соседние рабочие вершины, знаками (—), стоящими после цифры *3* и перед цифрой *2*. Теперь поиск работоспособной РК сводится к нахождению в табл. 8 такой строки,

Таблица 8

Вторичная таблица 1

№№ п/п	Номера рабочих вершин								Резервная вершина	Отказавшая вершина	Номера дуг				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
8_0	*8*	*9*	*0*	*1*	*3−*	*−2*	*6*	*7*	*5*	*4*	*7−8*	*8−1**	*0−2*	*3−7*	*6−9*
8_1	*9*	*0*	*1*	*3−*	*−2*	*6*	*7*	*5*	*8*	*4*	*5−9*	*9−3*	*1−6*	*2−5*	*7−0*
8_2	*0*	*1*	*3−*	*−2*	*6*	*7*	*5*	*8*	*9*	*4*	*8−0*	*0−2*	*3−7*	*6−8*	*5−1*
8_3	*1*	*3−*	*−2*	*6*	*7*	*5*	*8*	*9*	*0*	*4*	*9−1*	*1−6*	*2−5*	*7−9*	*8−3*
8_4	*3−*	*−2*	*6*	*7*	*5*	*8*	*9*	*0*	*1*	*4*	*0−3*	*3−7*	*6−8*	*5−0*	*9−2*
8_5	*−2*	*6*	*7*	*5*	*8*	*9*	*0*	*1*	*3−*	*4*	*1−2*	*2−5*	*7−9*	*8−1**	*0−6*
8_6	*6*	*7*	*5*	*8*	*9*	*0*	*1*	*3−*	*−2*	*4*	*3−6*	*6−8*	*5−0*	*9−3*	*1−7*
8_7	*7*	*5*	*8*	*9*	*0*	*1*	*3−*	*−2*	*6*	*4*	*2−7*	*7−9*	*8−1**	*0−2*	*3−5*
8_8	*5*	*8*	*9*	*0*	*1*	*3−*	*−2*	*6*	*7*	*4*	*6−5*	*5−0*	*9−3*	*1−6*	*2−8*

в которую не входят подозреваемые компоненты. Анализ записей табл. 8 показывает, что только одна строка с номером 8_6 удовлетворяет этим требованиям: в ней модуль *2* оказался в числе резервных, а линия (*8—1*) не входит в число рабочих.

После смены РК № 8 на РК № 8_6 получим такой гамильтонов цикл: $6 \to 7 \to 5 \to 8 \to 9 \to 0 \to \to 1 \to 3 \to 6$. С учетом этого цикла диагностический монитор модифицирует таблицы T_{S8} и T_{D8}, заменит номера рабочих модулей в соответствии с подстановкой (*67589013*) вместо (*89013267*), модифицирует таблицу состава и диагностическую таблицу, запустит процесс контроля РК № 8_6, который подтвердит ее работоспособность.

Для оценки эффективности реконфигурации 2-ОУ системы с целью устранения новых отказов в РК № 8_6 нужно построить вторичную таблицу на основании строки 8_6 из табл. 8. (Для сокращения объема статьи она здесь не приводится.) Анализ ее строк показывает, что из восьми РК только четыре работоспособны, но включают в себя одни и те же рабочие вершины и рабочие дуги. Другими словами, все эти РК соответствуют одной и той же неизбыточной восьмивершинной подсистеме, получающейся из 2-ОУ системы после удаления отказавших компонентов: модуля *4* и линий (*1—5*), (*5—1*), (*8—1*), (*1—8*), (*2—3*), (*3—2*). Следовательно, новый отказ любого компонента, входящего в состав РК № 8_6, приведет к отказу всей системы и потребует вмешательства человека-оператора для устранения накопившихся неисправностей и восстановления работоспособности исходной системы.

Сравнение состава отказавших компонентов данного примера с оценкой (1) позволяет заключить, что отказы внутренних дуг (*1—5*) и (*8—1*) не препятствовали устранению отказов модуля *4* и внешней дуги (*2—3*) и не понизили эту оценку.

ЗАКЛЮЧЕНИЕ

Разработан метод самодиагностирования возникающих отказов компонентов (модулей и линий связи) цифровых систем с реконфигурацией, для построения графов структур которых применены алгоритмы синтеза, предложенные М.Ф. Караваем. В основу метода положен принцип расширяющихся областей: выполнение необходимых процедур начинается с исправного модуля и последовательно передается другим модулям, исправность которых подтверждена предшествующими проверками. Допускаются устойчивые кратные отказы ограниченного числа компонентов, во время диагностирования могут возникать одиночные отказы. Для обнаружения возникающих отказов и получения правильного диагноза применяется дву- и более кратный запуск процесса самодиагностирования со сравнением результатов двух последних запусков.

Достоверность полученных результатов диагностирования основана на том, что все процедуры самодиагностирования исполняются модулями, исправность которых подтверждена на предыдущих этапах. В связи с этим наиболее "вредными" для предложенного метода самодиагностирования являются те неисправности, которые проявляются у модуля, признанного ранее исправным и получившего управление для выполнения диагностических функций. Результаты работы неисправного модуля могут быть самыми разными: начиная от полной остановки процесса самодиагностирования и кончая "хитрыми" искажениями информации, не обнаруживаемыми принятой схемой контроля. Так как в современных системах вероятность проявления подобных неисправностей в специально подобранные и сравнительно небольшие интервалы времени достаточно мала, то можно заключить, что предложенный метод самодиагностирования будет давать правильный диагноз для подавляющего числа неисправных компонентов. И только в единичных случаях для устранения возникшей неисправности потребуется помощь человека-оператора.

ЛИТЕРАТУРА

1. *Авиженис А.* Отказоустойчивость — свойство, обеспечивающее постоянную работоспособность цифровых систем // ТИИЭР. — 1978. — Т. 66. — № 10. — С. 5—25.
2. *Каравай М. Ф.* Инвариантно-групповой подход к исследованию *k*-отказоустойчивых структур // Автоматика и телемеханика. — 2000. — № 1. — С. 144—156.
3. *Maehle E., Moritzen K., Wirl K.* A graph model for diagnosis and reconfiguration and its application to a fault-tolerant multiprocessor system // 16-th Int. Conf. on Fault-Tolerant Computing FTCS-16, 1986. — P. 292—297.
4. *Ведешенков В. А.* Об использовании избыточных хорд *k*-отказоустойчивого графа для устранения влияния отказавших компонент // Автоматика и телемеханика. — 2003. — № 4. — С. 158—166.
5. *Ведешенков В. А.* Процедура восстановления работоспособности отказоустойчивых цифровых систем с динамической избыточностью // Автоматика и телемеханика. — 2003. — № 5. — С. 167—179.
6. *Ведешенков В. А.* Самодиагностирование отказов цифровых систем с реконфигу-рацией // Проблемы управления. — 2003. — № 4. — С. 39—51.
7. *Barsi F., Grandoni F., Maestrini P.* A theory of diagnosability of digital systems // IEEE Trans. Comput. — 1976. — Vol. C-25. — № 6. — P. 585—593.
8. *Preparata F. P., Metze G., Chien R. T.* On the connection assignment problem of diagnosable systems // IEEE Trans. Electr. Comput. — 1967. — Vol. 16. — № 6. — P. 848—854.
9. *Пархоменко П. П.* Определение технического состояния многопроцессорных вычис-лительных систем путем анализа графа синдромов // Автоматика и телемеханика. — 1999. — № 5. — С. 126−135.
10. *Ведешенков В. А.* Организация самодиагностирования технического состояния цифровых систем // Автоматика и телемеханика. — 2003. — № 11. — С. 165—182.
11. *Пархоменко П. П.* О системном диагностировании вершинных и реберных гипер-кубовых структур многопроцессорных вычислительных систем // Автоматика и телемеханика. — 1996. — № 11. — С. 165—174.

☎ *(095) 334-75-90*

E-mail: vva@l9.ipu.rssi.ru

❑

УДК 629.7.0172

ПРИНЦИПЫ ПОСТРОЕНИЯ БОРТОВЫХ КОМПЛЕКСОВ УПРАВЛЕНИЯ АВТОМАТИЧЕСКИХ КОСМИЧЕСКИХ АППАРАТОВ

Е. А. Микрин, Н. А. Суханов, В. Н. Платонов, И. В. Орловский, О. С. Котов, С. Г. Самсонов, В. Г. Беркут

Ракетно-космическая корпорация "Энергия" им. С.П. Королева, г. Королев

Изложен подход к построению бортовых комплексов управления (БКУ) автоматических космических аппаратов. Описаны основные компоненты БКУ и решаемые ими задачи. Приведена структура программного обеспечения — главного интеграционного звена БКУ. Рассмотрены характеристики БКУ космических аппаратов "Ямал", эксплуатируемых на геостационарной орбите в сфере телекоммуникаций.

Для обеспечения "жизнеспособности" разнообразных уже эксплуатируемых и только еще проектируемых космических аппаратов (КА) и выполнения ими целевых программ необходимо решить широкий круг задач, присущих всем типам КА. В числе этих задач следующие:

- обеспечение обмена информацией с наземным комплексом управления (НКУ);
- обеспечение снабжения аппарата электроэнергией;
- распределение электропитания на КА между потребителями;
- поддержание требуемого теплового режима на КА;
- определение и поддержание ориентации КА в пространстве;
- обеспечение движения КА в пространстве (перемещение его центра масс);
- обеспечение углового движения КА в пространстве (вокруг центра масс);
- определение (прогнозирование) местоположения КА на орбите;
- управление вращающимися солнечными батареями (при их наличии);
- сбор, хранение, обработка и передача телеметрической информации;
- управление работой систем и оборудования КА в соответствии с программой полета КА и с учетом его реального состояния.

При проектировании первых КА каждая задача решалась автономной работой отдельной системы, содержащей свою датчиковую аппаратуру, исполнительные органы, автоматику управления. С усложнением КА и увеличением числа решаемых ими задач появилась потребность в централизации управления и контроля за работой бортовых систем КА, прежде всего, в части рационального расходования и пополнения энергоресурсов, приоритетности и времени выполнения полетных и регламентных операций, автономного парирования нештатных ситуаций на основе результатов диагностики и тестирования бортовой аппаратуры и др. Разработка и внедрение на КА вычислительных средств с развитым программным обеспечением (ПО) позволили, в принципе, удовлетворить эту потребность. По аналогии с НКУ появилось понятие бортового комплекса управления (БКУ), объединяющего в себе основные бортовые системы КА и включающего в себя бортовую вычислительную систему, систему управления движением и навигацией, систему управления бортовой аппаратурой, бортовую аппаратуру служебного канала управления, систему бортовых измерений, а также программное обеспечение БКУ.

Для более детального рассмотрения принципов построения БКУ автоматических КА с целевой направленностью (связные спутники, КА наблюдения участков звездного пространства, космические аппараты зондирования Земли) перечислим их основные системы (рис. 1):

- бортовая вычислительная система в виде совокупности вычислительных средств и устройств сопряжения (адаптеров связи), обеспечивающая информационное взаимодействие с бортовыми

абонентами и предоставляющая свои вычислительные ресурсы для решения задач управления системами КА и задач контроля их работы;

- система управления движением и навигации или, в другой интерпретации, система ориентации и управления движением, предназначенная для управления движением КА как материальной точки (перемещением центра масс), так и для управления угловым движением КА (движением вокруг центра масс);
- система управления бортовой аппаратурой, выполняющая функции коммутации электропитания, усиления и преобразования электрических сигналов, а также выдачи команд управления в системы и приборы КА в соответствии с временными или логическими условиями;
- система бортовых измерений, предназначенная для сбора, обработки и передачи в НКУ телеметрической информации о результатах измерений, характеризующих состояние систем КА и протекающие на КА процессы;
- бортовая аппаратура служебного канала управления или командной радиолинии, представляющая собой радиотехнический комплекс для обеспечения своевременного обмена служебной информацией между НКУ и БКУ;
- объединенная двигательная установка, состоящая из комплекта двигателей для обеспечения перемещения КА относительно орбиты и углового движения КА;
- система обеспечения определенного теплового режима внутри КА;
- система энергоснабжения (СЭС) для преобразования первичной (солнечной) энергии в электрическую.

Задачи управления вращающимися солнечными батареями в некоторых типах КА решаются специальной системой ориентации солнечных батарей. В других КА эти задачи решаются в СУДН.

В некоторых классах КА в качестве отдельной структурной единицы рассматривается бортовая кабельная сеть.

Внедрение на КА вычислительных средств и новых конструктивно-технологических решений, применение современной элементной базы и средств комплексирования ПО позволили создать основу для построения интегрированных БКУ. Возможность оперативного контроля состояния систем КА и "умного" выполнения программы полета КА с учетом внешней обстановки, текущего статуса бортовых систем и имеющихся на текущий момент времени ресурсов позволила перенести многообразные функции контроля и управления КА в БВС, точнее — в ее ПО. Тенденция концентрации этих функций в бортовой вычислительной системе (ПО БКУ) продолжает усиливаться по мере развития программных и аппаратных средств. Программное обеспечение сформировалось как

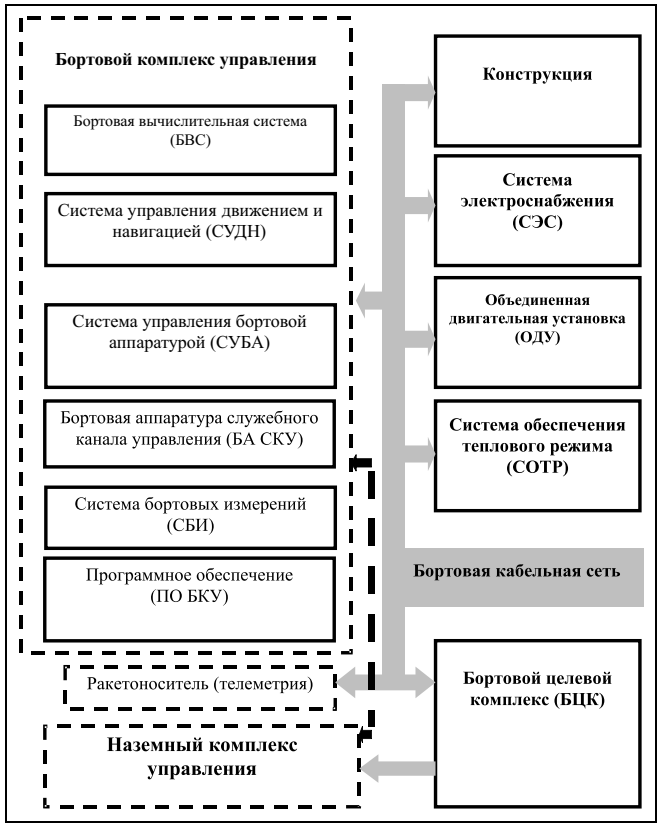


Рис. 1. Бортовые системы космического аппарата

отдельный (и одним из самых главных) компонентов БКУ (рис. 2).

Программное обеспечение БКУ построено по иерархическому принципу:

- первый или нижний уровень составляют драйверы обмена с аппаратурой и программы организации вычислительного процесса;
- второй уровень составляют программы обеспечения управления и контроля работы бортовых приборов и оборудования;
- программы третьего уровня включают в себя программы обеспечения полетных режимов бортовых систем и расчетные программы;
- четвертый или верхний уровень составляют программы планирования и организации режимов работы всего БКУ и контроль состояния систем КА.

Архитектура ПО БКУ подразделяет все программы на служебные (диспетчер, обмен, управление конфигурацией БВС, таймирование и др.) и функциональные (программы включения/выключения конкретных приборов, программы расчета различной подготовительной и сопроводительной информации, программы формирования управляющих воздействий на отдельные приборы и т. п.). Каждая программа (программный модуль) имеет свои настроечные параметры и логико-информа-

Рис. 2. Структура программного обеспечения

ционные связи с другими программами. Построение ПО БКУ предполагает детерминированное циркулирование обменной информации между программами всех уровней, причем управляющая информация поступает сверху вниз (от программ верхних уровней до программ нижних уровней), а контрольно-диагностическая информация — снизу вверх. Для обеспечения функционирования ПО БКУ в реальном масштабе времени каждой программе определяются последовательность и конкретное время подключения на вычислительном такте бортового компьютера, а также ее вычислительные ресурсы.

Интеграционный характер ПО БКУ позволяет выполнять, кроме функций контроля и управления, и другие важные функции и задачи, как, например:

- задачи повиткового планирования полета КА;
- задачи оптимизации расхода бортовых ресурсов;
- функции обеспечения автономности существования КА;
- функции оперативного реагирования на многие нештатные ситуации.

Получение цифровой информации от бортовых систем (напрямую или через СБИ), обмен информацией с НКУ (через БА СКУ), обработка и использование полученной информации в расчетно-вычислительных задачах, реализованных в ПО, — все это выдвинуло БВС и ПО БКУ на главенствующее место в бортовом комплексе управления. Системы БА СКУ и СБИ в такой конфигурации являются неотъемлемой частью БКУ как источники циркулирующей на КА информации и необходимые звенья, поддерживающие информационно-логические интерфейсы. Заметим, что связи между БВС, БА СКУ и СБИ как физические (проводные, через бортовую кабельную сеть и устройства сопряжения), так и “виртуальные” (информационные, через каналы информационного обмена).

Неотъемлемой и основной частью БКУ следует считать и систему управления бортовой аппаратурой. Две важные функции СУБА носят интеграционный характер и являются прерогативой БКУ:

- обеспечение всех бортовых потребителей электропитанием;
- обеспечение физического (проводного) интерфейса с системами и оборудованием КА и управление ими путем формирования соответствующих команд и сигналов.

Все остальные из вышеперечисленных систем решают свои конкретные задачи, жизненно важные для КА, но не являющиеся интеграционными с точки зрения структурного построения БКУ. Из этих систем особо выделим СУДН, часто включаемую разработчиками КА в состав БКУ благодаря следующим аспектам:

- данная система (как и СУБА) — одна из первых бортовых систем, она проектировалась и разрабатывалась уже для первых КА;
- задачи СУДН (ориентация, стабилизация, наведение КА для решения целевых задач и т. д.) — важнейшие и первоочередные;
- программы управления СУДН тесно “привязаны” к программам управления других систем и программам “верхнего” уровня ПО БКУ и др.

В состав СУДН включены чувствительные элементы в виде оптико-спектральных датчиков и датчиков угловых скоростей, преобразующие устройства и блоки формирования управляющих сигналов, а также исполнительные органы в виде силовых гироприборов (например, маховиков или гиродинов). Исполнительными органами СУДН служат также двигатели двигательной установки. Состав аппаратуры СУДН может дополняться навигационными приборами и аппаратурой спутниковой навигации.

Основой построения СУДН разработки РКК “Энергия” является корректируемая бесплатформенная инерциальная навигационная система, позволяющая в реальном масштабе времени определять текущее положение связанных осей КА отно-

сительно инерциальной системы координат путем интегрирования составляющих абсолютной угловой скорости (рис. 3).

Функционирование СУДН обеспечивается работой трех контуров управления:

- кинематический контур управления ориентацией позволяет определять угловое рассогласование приборного базиса СУДН с неким опорным базисом, задаваемым режимом ориентации; кроме того, кинематический контур уточняет (корректирует) данный опорный базис;
- динамический контур управления ориентацией позволяет совмещать с заданной точностью связанный базис КА с опорным базисом; кроме того, динамический контур обеспечивает стабилизацию КА в процессе коррекции орбиты и других динамических операций;
- навигационный контур позволяет определять (прогнозировать) в реальном масштабе времени с заданной точностью местоположение КА на орбите по начальным условиям, определяемым в НКУ или в бортовой навигационной аппаратуре, с использованием расчетных гравитационных и магнитных моделей Земли и информации от бортовых навигационных датчиков КА.

К служебным системам автоматических КА рассматриваемого класса предъявляются сходные по своему составу требования, обусловленные полезной нагрузкой (или БЦК) к служебным системам этих аппаратов. В головных предприятиях космической отрасли наметилась тенденция разработки универсальных космических платформ (УКП), единых для различных типов автоматических КА. Интеграция готовой УКП с новым БЦК путем преемственности систем и приборов УКП и минимизации этапов проектирования и наземной отработки позволяет существенно сократить сроки создания новых КА и финансовые затраты при сохранении показателей качества и надежности.

Степень преемственности существующей космической платформы к применению на КА с новой целевой аппаратурой определяется, исходя из характеристик этой аппаратуры и состава требований к УКП, ее обусловленных. Основные характеристики БЦК: энергопотребление, масса, количество фидеров питания, тип интерфейсов и др. В число задач, определяемых условиями работы БЦК, включаются:

- обеспечение точности ориентации и стабилизации КА при работе БЦК;
- обеспечение точности наведения и ее оценка;
- обеспечение заданных параметров рабочей орбиты и длительности эксплуатации КА;
- выполнение заданных режимов работы БЦК и др.

Следствием анализа требований и характеристик БЦК является подтверждение пригодности УКП к интеграции с данным БЦК в составе проектируемого КА либо определение перечня необходимых доработок и дооснащений УКП. Выбор окончательной модификации УКП применительно к новому БЦК определяется как решение многопараметрического функционала:

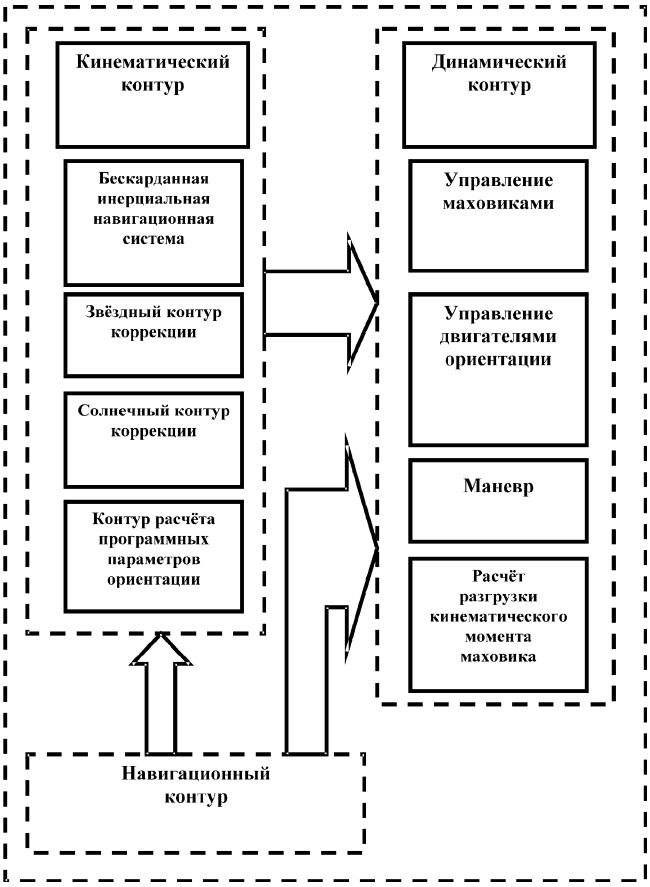


Рис. 3. Структура СУДН на базе двухконтурной бесплатформенной инерциальной системы

$$\Phi\,(M_i,\ N_{ij},\ U_i,\ T_i,\ S_i,\ \ldots,\ K_{ij},\ P_{ij}) = 0,$$

где M_i — массы отдельных приборов и систем; N_{ij} — число однотипных приборов j-го класса в i-й системе; U_i — энергопотребление отдельных приборов и систем; K_{ij} — число интерфейсов (командных сигналов) j-го типа в i-й системе; P_{ij} — вероятность безотказной работы прибора j-го типа в i-й системе; T_i — длительность изготовления и испытаний отдельных компонентов; S_i — стоимость затрат на создание отдельных компонентов.

В такой схеме БКУ рассматривается как ядро УКП. Другими словами, БКУ является совокупностью основных служебных систем и бортового ПО, обеспечивающей интеграцию систем УКП при реализации функциональных задач КА на программно-логическом и физическом уровнях через

соответствующие интерфейсы. Основные задачи БКУ:

- координированное управление функционированием УКП и КА в целом при наземной отработке КА и штатном выполнении программы полета в автоматическом режиме и по информации от НКУ;
- диагностика состояния УКП и ее систем, обнаружение, локализация и парирование расчетных нештатных ситуаций в автоматическом режиме;
- сбор, первичная обработка, хранение и телеметрирование информации оперативного контроля, а также ее использование в задачах управления УКП;
- организация информационно-командного взаимодействия с БЦК.

Решение бортовым комплексом управления перечисленных задач сопровождается подтверждением качественных характеристик, таких как:

- высокий уровень комплексирования, т. е. организация комбинированной работы систем УКП и поддержка интерфейсов при реализации всех полетных режимов и операций в реальном масштабе времени;
- строгая иерархичность многоуровневой структуры построения БКУ и детерминированность движения потоков информации (командно-управляющей информации сверху вниз от "ядра" БКУ к каждому элементу и контрольно-диагностической информации снизу вверх от периферийных элементов до "ядра");
- развитая "гибкость" управления, реализованная в алгоритмах управления ПО БКУ и обеспечивающая на базе целеуказаний от НКУ и обработанной диагностической информации от датчиковой аппаратуры эффективный расход ресурсов и задействование резервов, а также адаптивность к отказам и парирование нештатных ситуаций.

В РКК "Энергия" универсальная космическая платформа для связных и исследовательских КА была разработана в рамках тематики спутниковой системы "Ямал". Основная целевая функция КА "Ямал" заключается в предоставлении пользователям спутниковой системы связи "Ямал" услуг в части телекоммуникаций. Запуск в сентябре 1999 г. КА "Ямал-100" и в ноябре 2003 г. двух аппаратов "Ямал-200", а также последующая эксплуатация этих КА подтвердили идеологическую правильность выбранного направления в построении бортовых комплексов управления.

Бортовой комплекс управления универсальной платформы "Ямал" имеет следующие характеристики:

- троированное резервирование аппаратуры БВС; емкость памяти одной ЦВМ 1024 Кбайт, тактовая частота центральных процессоров 12 МГц;
- распределяемая мощность до 10 кВт, более 150 фидеров питания, около 600 команд управления;
- скорость приема информации из НКУ — 1 Кбит/с, скорость передачи информации на НКУ (при ориентированном полете КА) — 4,8 Кбит/с;
- измеряемых параметров — более 1000 дискретных, температурных и аналоговых;
- вероятность безотказной работы на орбите в течение 10 лет — не менее 0,945.

Реализованный вариант БКУ для УКП "Ямал" позволяет обеспечить требования к КА данной системы, основные из которых заключаются в:

- высокой надежности и длительности непрерывной эксплуатации на геостационарной орбите (ГСО);
- максимальной автономности функционирования КА;
- точности удержания КА в рабочей точке на ГСО 0,1°;
- точности наведения и удержания осей связной аппаратуры 0,2°;
- погрешности определения навигационных параметров по координатам не более 10 км.

Отработанная и готовая к применению в составе целого класса космических аппаратов платформа создает особенную привлекательность для заказчиков перспективных космических изделий. Развитие и совершенствование БКУ предполагается в следующих направлениях:

- доработка (замена на новые разработки) отдельных приборов и исполнительных органов БКУ, позволяющих улучшить точностные, динамические, массово-инерционные характеристики из условия требований со стороны целевой аппаратуры;
- повышение надежности и увеличение гарантированного срока функционирования отдельных приборов (путем применения современных электрорадиоизделий) и БКУ в целом (благодаря внедрению функционального резервирования на аппаратном и программном уровнях);
- при сохранении целевых функций и показателей БКУ снижение его общей массы и экономия интерфейсных ресурсов в интересах полезной нагрузки благодаря унификации разнотипных интерфейсов, применению более легких электрорадиоизделий, оптимизации размещения бортовой кабельной сети, переходу в отдельных позициях структуры БКУ на дублирование вместо троирования аппаратных средств.

Каждое направление предполагает выполнение комплекса работ и мероприятий и требует отдельного рассмотрения.

☎ *(095) 513-63-46*

E-mail: Eugeny.Mikrin@rsce.ru ❑

УДК 519.712

МУЛЬТИПЛИКАТИВНАЯ АЛГЕБРА И ЕЕ ПРИМЕНЕНИЕ В ЛОГИЧЕСКОЙ ОБРАБОТКЕ ДАННЫХ

В. С. Выхованец, В. Д. Малюгин

Институт проблем управления им. В. А. Трапезникова, г. Москва

Рассмотрена проблема синтеза формул для дискретных функций на основе многоступенчатой (решающей) и спектральной декомпозиции, выполняемой в мультипликативной алгебре. Показана связь синтеза мультипликативных форм с общей задачей синтеза формул. Приведены практические рекомендации по программной и аппаратной реализации получаемых формул.

ВВЕДЕНИЕ

Логическая обработка данных широко применяется в логическом управлении, цифровой обработке сигналов, распознавании образов, синтезе дискретных устройств и в других областях. В свою очередь, задачи логического управления возникают в дискретных системах управления, которые используют дискретные сигналы и функционируют в дискретном времени (рис. 1).

Система управления дискретного действия может быть описана совокупностью логических функций (системой функций). В результате представления этих функций в одной из форм их описания, например, в виде выражений в некотором функционально полном базисе операций, получают логическую модель объекта (среды), состоящую из отдельных описаний каждой функции.

Будем исходить из того, что описание всякой функции нужно для ее вычисления (нахождения ее значений) на заданных входных данных.

Данные при логической обработке представляются в виде конечных последовательностей знаков (цифр), взятых из некоторого конечного алфавита N, а сама обработка осуществляется путем преобразования входных данных $\mathbf{X}$ в выходные $\mathbf{Y}$ посредством разделения входных данных $\mathbf{X}$ на части (переменные) $x_0, ..., x_{n-1}$, выполнения над ними некоторых последовательностей операций $\mathbf{F} = \{f_0, ..., f_{s-1}\}$ с целью получения результатов $y_0, ..., y_{n-1}$, из которых потом формируются выходные данные $\mathbf{Y}$ (рис. 2).

Основная проблема логической обработки состоит в декомпозиции дискретной функции или ее представлении в виде формулы как некоторой последовательности операций над переменными.

Попытки найти декомпозицию функции общего вида (нерегулярную декомпозицию) столкнулись с серьезными трудностями, связанными с предельно общей постановкой задачи, и при больших размерностях — практически нереализуемой [1]. Нерегулярная декомпозиция сводится к

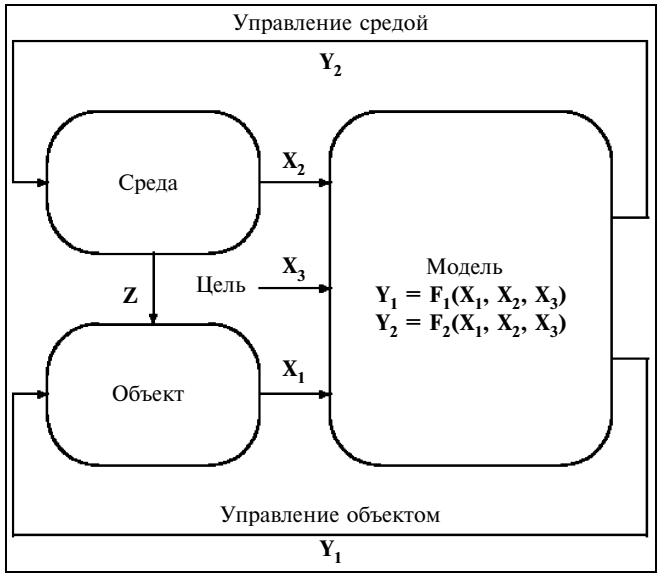


Рис. 1. Схема логического управления

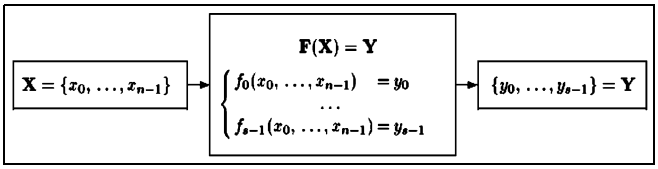


Рис. 2. Схема логической обработки данных

NP-полной задаче раскраски графа, для которой не найдено эффективных решений [2]. Поэтому нерегулярную обработку данных представляют, в основном, в алгоритмических формах. Теоретической базой при этом служат хорошо известные парадигмы анализа предметной области, получившие свою реализацию в одноименных методологиях проектирования программных средств (логической, функциональной, структурной, объектно-ориентированной, и др.).

С другой стороны, поиск методов дискретной декомпозиции и дискретной минимизации пошел в направлении исследования регулярных аналитических конструкций, предельным случаем которых являются спектральные разложения. Традиционно первым методом компактного представления логической модели стала индивидуальная минимизация функций, которая заключается в поиске минимальных или близких к минимальным форм для каждой функции системы в отдельности. Очевидно, при этом в процессе преобразования общее количество формул остается неизменным, представления функций упрощаются, но приходится жертвовать единым базисом, так как каждая функция имеет свой в некотором смысле оптимальный набор операций для своего наиболее простого выражения.

Другой метод получения и упрощения логической модели заключается в совместной минимизации функций системы, при которой выделяются некоторые общие подвыражения, участвующие при вычислении двух и более функций. В итоге количество функций системы увеличивается, но при аппаратурной реализации уменьшается требуемое количество логических элементов, а при программной реализации — общее время вычисления модели.

Как для первого, так и для этого метода пока не найдено эффективных алгоритмов, позволяющих за приемлемое время минимизировать достаточно сложную модель. Как правило, применяются переборные алгоритмы, имеющие высокую трудоемкость. Так как индивидуальная и совместная минимизации функций сохраняют их общее число неизменным или даже увеличивает, то для повышения эффективности представления модели ставится задача поиска форм совместного описания нескольких функций, например, в виде единого выражения, позволяющего вычислять все функции сразу, не вычисляя каждую в отдельности.

Известно, что произвольная функция может быть представлена со сложностью, не превышающей некоторую максимальную, не зависящую от используемого базиса. Поэтому целесообразно поставить задачу поиска конечных форм совместного описания всей системы функций, например, в виде единого выражения в некотором общем для всех функций базисе. Хотя сложность этого выражения может быть максимальной, зато будет описываться вся система сразу и функции системы будут вычисляться одновременно (параллельно) при вычислении этого выражения. Это позволяет не только уменьшить объем памяти, необходимый для хранения модели, но и сократить время ее вычисления.

Одной из первых и самой распространенной формой совместного описания системы функций алгебры логики является арифметический полином [3]. Предельный случай упрощения логической модели состоит в ее представлении в виде линейного выражения — суммы переменных, умноженных на некоторые коэффициенты, причем операции сложения и умножения не обязательно должны быть традиционными арифметическими. Ранее показано [4], что из-за увеличения разрядности коэффициентов произвольная система булевых функций может быть представлена в виде композиции двух линейных арифметических полиномов ограниченной сложности. Если даже выражение нелинейное, то специальное увеличение разрядности коэффициентов позволяет реализовать кратные логические вычисления, при которых система функций вычисляется сразу (одновременно, параллельно) на нескольких наборах переменных [5]. Заметим, что в англоязычных публикациях структурированное объединение логических выражений называется представлением функций в форме слов (word-level).

Все перечисленные методы основаны на спектральной декомпозиции, при которой функция разлагается в линейную комбинацию фиксированного множества других функций, называемых спектральными и имеющими такую же длину характеристических векторов, что и исходная. С технической точки зрения такой подход оправдан, когда эффективность реализации спектрального базиса достаточно высока. Однородность спектральных разложений не позволяет гарантировать эффективное формульное представление наперед неизвестной функции. Поэтому развитие спектральных методов связано с уменьшением регулярности синтезируемых формул, в частности, с применением многоступенчатой декомпозиции (решающих диаграмм) [6, 7], с объединением спектральных составляющих в различных арифметиках [8—11],

совместного описания множества функций в виде одного спектра [3], применения новых и расширения имеющихся спектральных базисов [12—16].

Интерес к спектральным методам логики и распараллеливанию логических вычислений вызван появлением особо сложных задач, в которых многочисленные данные связаны логическойзависимостью. При этом реализация логических алгоритмов обычно мыслиться как программная. Принятое ныне описание схем и алгоритмов, основанное на языке булевой алгебры, лишь частично соответствует программным и техническим возможностям современных ЭВМ.

Так как в основе описанных подходов лежат способы совместного описания системы логических функций посредством какой-нибудь единой формулы, то появляется надежда, что сведение системы функций к одному выражению упрощает и процедуру проверки устройства, реализующего эту формулу. Иначе говоря, упрощается тестирование, диагностирование устройств и верификация соответствующих программ.

В настоящей работе рассмотрено применение мультипликативной алгебры для многоступенчатой (решающей) и спектральной декомпозиции. В отличие от традиционного подхода, когда объединение спектральных составляющих осуществляется в таких алгебраических системах как конечное поле или кольцо целых чисел, предлагается применение специальной алгебраической системы — мультипликативной алгебры, позволяющей оценить сложность представления функции заранее, до проведения декомпозиции, снизить трудоемкость синтеза формульного выражения и уменьшить число операций, необходимых для вычисления (реализации) функции.

В §1 даются необходимые определения и вводятся основные понятия, которые поясняются на примерах. Современному состоянию проблемы представления дискретных функций в виде формул посвящен §2, где рассмотрена алгебраическая декомпозиция, осуществляемая в различных образующих алгебрах. Строгое определение мультипликативной алгебры и основные теоремы разложения приведены в §3. Заключительный раздел статьи посвящен аппаратной и программной реализации логических моделей в мультипликативных формах.

1. ОПРЕДЕЛЕНИЯ

В логической обработке применяется дискретное кодирование данных. Выберем в качестве универсального множество целых чисел $N_0 = \{..., -1, 0, 1, ...\}$. Когда необходимо задать конечное множество, будем пользоваться подмножеством N_k этого множества, $N_k = \{0, 1, ..., k-1\}$, где $k > 0$ — число элементов в нем.

1.1. Дискретные функции

Функцией f одной переменной x, заданной на множестве X и принимающей значения на множестве Y, называется отношение между X и Y такое, что каждый элемент $x \in X$ связан не более, чем с одним элементом $y \in Y$. Множество X называется областью определения функции, множество Y — областью значений. Функцию будем называть дискретной, если ее область определения и область значений — конечные множества.

Пусть функция f задана на множестве $N_{k_0} \times \times N_{k_1} \times ... \times N_{k_{n-1}}$ и принимает значения на множестве N_{k_f} (здесь знаком × обозначено декартово произведение множеств). Это означает, что функция f зависит от n переменных $x_0, x_1, ..., x_{n-1}$, т. е. $y = f(x_0, x_1, ..., x_{n-1})$, где $y \in N_{k_f}$ и $x_j \in N_{k_j}$ ($j = \overline{0, n-1}$). Будем говорить, что функция имеет значность k_f, а переменная x_j — значность k_j.

Дискретную функцию определим таблицей истинности, содержащей $m = k_0 k_1 ... k_{n-1}$ строк, где m — число элементов области определения (табл. 1). Установим взаимно однозначное соответствие между значением i переменной $x \in N_m$ и значениями i_j переменных $x_j \in N_{k_j}$ с помощью представления числа i в n-разрядной позиционной системе счисления с основаниями, определяемыми значностями переменных,

$$i = (i_0, i_1, ..., i_{n-1})_{k_0 k_1 ... k_{n-1}}, \qquad (1)$$

где $i_j \in N_{k_j}$ — j-я цифра числа i, представленного в системе счисления с основаниями $k_0, k_1, ..., k_{n-1}$. Будем полагать, что цифра i_0 имеет наименьший вес.

Таблица 1

Таблица истинности функции

x	x_0	x_1	...	x_{n-1}	y
0	0	0	...	0	y_0
1	1	0	...	0	y_1
...	...	...	...	...	...
$k_0 - 1$	$k_0 - 1$	0	...	0	y_{k_0-1}
...	...	...	...	...	...
i	i_0	i_1	...	i_{n-1}	y_i
...	...	...	...	...	...
$m - 1$	$k_0 - 1$	$k_1 - 1$	...	$k_{n-1} - 1$	y_{m-1}

Для прямого и обратного преобразования числа i в его представлении (1) будем пользоваться следующие рекуррентными выражениями:

$$k = k_0 k_1 ... k_{n-1}, \quad \left.\begin{array}{l} i_j = i \bmod k, \\ i = i \operatorname{div} k, \\ k = k / k_{j-1}, \end{array}\right\} \quad (j = \overline{n-1, 1})$$

— для прямого преобразования и

$$k = 1,\ i = 0, \quad \left.\begin{array}{l} i = i + i_j k, \\ k = k k_{j-1}, \end{array}\right\} \quad (j = \overline{0, n-1})$$

— для обратного, где mod обозначает остаток, а div — целую часть от деления первого операнда на второй.

Вектором будем называть упорядоченное множество элементов, перечисленное в квадратных скобках (для отличия от представления числа в системе счисления (1), где используются круглые скобки). Как и ранее, будем предполагать, что первый элемент вектора имеет наименьшую значимость. Произвольная функция может быть задана вектором значений или характеристическим вектором $\mathbf{F} = [y_0\, y_1 ...\, y_{m-1}]$ длины m и вектором значностей переменных $\mathbf{K} = [k_0\ k_1\ ...\ k_{n-1}]$ длины n. Если функция задана только вектором **F**, то существует множество векторов **K**, которыми можно воспользоваться для полного определения функции. Число таких векторов равно числу представлений числа m в виде произведения целых чисел, больших единицы.

Вектор переменных обозначим как $\mathbf{X} = [x_0\ x_1\ ...\ x_{n-1}]$. Выражение $\mathbf{F} = [x_0\ x_1\ ...\ x_{n-1}]$ равно значению функции, заданной характеристическим вектором, когда переменные принимают значения x_0, x_1, ..., x_{n-1}. Возможны также записи вида $\mathbf{F}(x)$, $\mathbf{F}(x_0, x_1, ..., x_{n-1})$, $\mathbf{F}(x_0, x_1, ..., x_{n-1})_{k_0 k_1 ... k_{n-1}}$, если переменная x представлена в позиционной системе счисления с известными основаниями **K**, а $(x_0, x_1, ..., x_{n-1})$ читается как последовательность цифр числа x в этой системе счисления.

Таблица 2

Дискретная функция

$\mathbf{K}_0 = [6]$	$\mathbf{K}_1 = [32]$		$\mathbf{K}_2 = [23]$		$k_f = 4$
x	x_0	x_1	x_0	x_1	f
0	0	0	0	0	3
1	1	0	1	0	0
2	2	0	0	1	1
3	0	1	1	1	2
4	1	1	0	2	2
5	2	1	1	2	0

Пример 1. Пусть дискретная функция задана характеристическим вектором **F** = [301220] длины 6. Функция имеет значность $k_f = 4$, так как элементы вектора **F** принимают значения на множестве из четырех элементов $N_4 = \{0, 1, 2, 3\}$. Для вектора **K** возможны три варианта значностей переменных: $\mathbf{K_0} = [6]$, $\mathbf{K_0} = [3\ 2]$ и $\mathbf{K_0} = [2\ 3]$. Все три функции приведены в табл. 2.

1.2. Дискретные операции

Одна и та же логическая обработка данных (дискретная функция) может иметь несколько представлений, которые различаются разделением входных данных на части (переменные). Входные данные преобразуются в выходные (значения вычисляемых функций) путем выполнения унарных, бинарных, тернарных, и т. д., r-арных операций. Арность или местность операции определяется числом операндов (переменных), участвующих в формировании результата операции. Очевидно, если результат r-арной операции не зависит от одного из операндов, такую операцию следует рассматривать как $(r-1)$-арную. Таким образом, операцией следует назвать функцию, существенно зависящую от своих переменных.

Традиционно, в логической обработке данных, используются унарные и бинарные дискретные операции. Унарная операция может быть определена вектором, а бинарная — матрицей. Вектор $[y_0 y_1 ... y_{k_0-1}]$ длины k_0, обозначаемый также как $[y_i]$, является вектором унарной операции, если он содержит, как минимум, один элемент, отличающийся от других. Единственный операнд этой операции имеет значность k_0. Матрица размерности $k_0 \times k_1$, обозначаемая как $[y_{ij}]$, является матрицей бинарной операции, если все ее строки (столбцы) не одинаковы. В этом случае первый операнд имеет значность k_0, а второй — k_1. Будем обозначать применение унарной операции как $[y_i](x_0) = y_{x_0}$, а бинарной — $[y_{ij}](x_0, x_1) = y_{x_0 x_1}$ или $x_0[y_{ij}]x_1 = y_{x_0 x_1}$.

Пример 2. Рассмотрим дискретные функции из табл. 2. Эти функции задают три четырехзначные операции

$$\begin{bmatrix} 3 \\ 0 \\ 1 \\ 2 \\ 2 \\ 0 \end{bmatrix} (5) = 0, \quad \begin{bmatrix} 3 & 0 \\ 1 & 2 \\ 2 & 0 \end{bmatrix} (2, 1) = 0, \quad \begin{bmatrix} 3 & 0 & 1 \\ 2 & 2 & 0 \end{bmatrix} (1, 2) = 0,$$

Например, для второй из этих операций вычисляется значение, равное нулю при $x_0 = 2$ и $x_1 = 1$.

2. АЛГЕБРАИЧЕСКАЯ ДЕКОМПОЗИЦИЯ

Основная задача логической обработки состоит в декомпозиции дискретной функции или ее представлении в виде формулы (выражения), связывающего переменные и операции над ними со значением функции. Известны два крайних подхода к построению формульных представлений. В соответствии с одним из них входные переменные преобразуются в значение функции на основе неалгебраических представлений. Другой подход основан на представлении функции в регулярных (алгебраических) формах. Спектральные разложения, в свою очередь, являются предельным случаем алгебраического представления. В табл. 3 приведены основные виды дискретной декомпозиции.

Пересекающаяся [17], разделительная [18, 19] и кратная [20] декомпозиции относятся к классу неалгебраических. Решающая [21] и спектральная [22, 23] выполняются в некоторой алгебраической системе с двумя бинарными операциями и относятся к классу алгебраических. В свою очередь, промежуточная декомпозиция [24] является прообразом алгебраической и располагается между алгебраическими и неалгебраическими.

2.1. Разделение переменных

Декомпозиции дискретных функций, кроме пересекающейся, основаны на разделении переменных. Для этого множество переменных $X = \{x_0, x_1, ..., x_{n-1}\}$ разбивается на два непересекающихся подмножества $X' = \{x'_0, x'_1, ..., x'_{n'-1}\}$ и $X'' = \{x''_0, x''_1, ..., x''_{n''-1}\}$ с числом элементов n' и n'' и значностями $k' = \prod_{j=0}^{n'-1} k'_j$ и $k'' = \prod_{j=0}^{n''-1} k''_j$ или, что то же самое, функцию рассматривают зависящей от двух переменных x' и x'' со значностями k' и k'', где $k'k'' = m$. Поясним разделение переменных на примере.

Пример 3. Пусть функция определена векторами

$$\mathbf{F} = [2010321210301010321200 10], \quad \mathbf{K} = [2232],$$

т. е. зависит от переменных $X = \{x_0, x_1, x_2, x_3\}$ со значностями $k_0 = k_1 = k_3 = 2$ и $k_2 = 3$. Разобьем переменные на два подмножества $X' = \{x_0, x_2\}$ и $X'' = \{x_1, x_3\}$. Тогда $k' = k_0k_2 = 6$, а $k'' = k_1k_3 = 4$. Переопределим функцию в виде двумерной таблицы (табл. 4).

При построении табл. 4 переменная $x \in N_{24}$ пересчитывалась в переменные $x' \in N_6$ и $x'' \in N_4$ с использованием позиционной системы счисления с основаниями $\mathbf{K}' = [23]$ и $\mathbf{K}'' = [22]$ так, что

$$\mathbf{F}(x) = \mathbf{F}(x_0, x_1, x_2, x_3)_{2232} = \mathbf{T}[(x_0, x_2)_{22}(x_1, x_3)_{23}] = \mathbf{T}[c, r] = t_{rc},$$

где $\mathbf{T}$ — матрица функции, t_{rc} — элемент этой матрицы, а $r(c)$ — ее строка (столбец). В частности, $t_{32} = \mathbf{T}[23] = \mathbf{T}[(0, 1)_{22}(1, 1)_{23}] = \mathbf{F}[(0111)_{2232}] = \mathbf{F}[18] = 2$.

Заметим, что число способов, которыми множество из n переменных может быть разделено на два непересекающихся подмножества, равно $2^{n-1} - 1$.

2.2. Разложение функции

Решающая и спектральная декомпозиции выполняются в некоторой алгебре $R = \langle N_k, +, \times \rangle$ (алгебре образующих операций), заданной на множестве N_k двумя бинарными операциями, условно называемыми сложением и умножением. Наибольшее распространение на практике получили такие образующие алгебры, как конечное поле ($k = p^q$, где p — простое, q — натуральное) и кольцо целых чисел ($k = 0$).

Таблица 3

Декомпозиции дискретной функции

Декопозиция	Конструкция
Пересекающаяся	$f(X) = \theta(X''\, a(X'))$, $X' \cap X'' \neq 0$
Разделительная	$f(X) = \theta(X''\, a(X'))$, $X' \cap X'' = 0$
Кратная	$f(X) = \theta(X''\, a_1(X''))$, ..., $a_m(X'')$
Промежуточная	$f(X) = \sigma(\theta_1(X', a_1(X'')), ..., \theta_m(X', a_m(X'')$
Решающая	$f(X) = \sum_{i=1}^{m} \theta_i(X') \times a_i(X'')$
Спектральная	$f(X) = \sum_{i=1}^{m} \theta_i(X') \times a_i$

Таблица 4

Двумерная таблица дискретной функции

x_0	x_2	x_1	0	1	0	1
		x_3	0	0	1	1
		$X'\backslash X''$	0	1	2	3
0	0	0	2	1	1	1
1	0	1	0	0	0	0
0	1	2	3	1	3	1
1	1	3	2	2	2	2
0	2	4	1	3	0	1
1	2	5	0	0	0	0

Пусть задана произвольная m-функция f (функция с длиной характеристического вектора m) значности $k_f < k$. Представим эту функцию в виде разложения по системе k'-функций θ_i ($i = \overline{0, k'-1}$),

$$f(X) = \sum_{i=0}^{k'-1} \theta_i(X') \times a_i(X''), \qquad (2)$$

где a_i — коэффициенты разложения (некоторые k''-функции). Будем предполагать, что спектральные функции θ_i известны. Тогда для нахождения коэффициентов разложения необходимо решить систему линейных уравнений

$$\sum_{i=0}^{k'-1} d_{ji} \times a_i(X'') = f(j, X'') \quad (i = \overline{0, k'-1}) \qquad (3)$$

относительно неизвестных функций $a_i(X'')$, где d_{ji} — некоторые константы, $d_{ji} = \theta_i(j)$. Если формульные представления спектральных функций заданы, то после решения системы уравнений (2) получим выражение для функции f, в котором участвуют вычисленные функции-коэффициенты a_i, представленные в табличной форме.

Функции a_i будем раскладывать до тех пор, пока в качестве коэффициентов не появятся константы. В итоге получим формульное представление для исходной функции f. Декомпозиция (2) может быть записана также в матричном виде

$$\mathbf{T} = \mathbf{D} \times \mathbf{A} \quad (\mathbf{A} = \mathbf{Q} \times \mathbf{T}), \qquad (4)$$

где **T** (**A**) — матрица функции (коэффициентов) размерности $k' \times k''$, **D** (**Q**) — матрица прямого (обратного) преобразования размерности $k' \times k'$ с элементами, принимающими значения на множестве N_k. Заметим, что столбцы матрицы **D** являются характеристическими векторами спектральных функций θ_i. Поясним сказанное на примере.

Пример 4. Разложим дискретную функцию в конечном поле, образованном операциями

$$+ = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 3 & 2 \\ 2 & 3 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{bmatrix}, \quad \times = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 2 & 3 & 1 \\ 0 & 3 & 1 & 2 \\ 0 & 1 & 2 & 3 \end{bmatrix},$$

заданными на множестве N_4 матрицами Кэли. Нулем этого поля является 0, а единицей — 3. Рассмотрим функцию из табл. 2, когда $X' = \{x_0\}$ и $X'' = \{x_1\}$ при $k' = 3$ и $k'' = 2$. После разделения переменных получим двумерную таблицу:

$X' \backslash X''$	0	1
0	3	2
1	0	2
2	1	0

Из выражения (2) следует, что

$$f(X) = \theta_0(X') \times a_0(X'') + \theta_1(X') \times a_1(X'') + \theta_2(X') \times a_2(X'').$$

Определим систему спектральных функций так: $\theta_0 = 3$, $\theta_0 = \hat{x}_0$, $\theta_0 = \tilde{x}_0$, где $\hat{x}$ — увеличение x на единицу, а $\tilde{x}$ — сдвиг x вправо на один двоичный разряд.

Построим матрицу прямого преобразования **D** и найдем в заданном поле обратную ей матрицу **Q**:

$$\mathbf{D} = \begin{bmatrix} 3 & 1 & 0 \\ 3 & 2 & 0 \\ 3 & 3 & 1 \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} 2 & 1 & 0 \\ 3 & 3 & 0 \\ 3 & 1 & 2 \end{bmatrix}.$$

Далее вычислим матрицу коэффициентов

$$\mathbf{A} = \mathbf{Q} \times \mathbf{T} = \begin{bmatrix} 2 & 1 & 0 \\ 3 & 3 & 0 \\ 3 & 1 & 2 \end{bmatrix} \times \begin{bmatrix} 3 & 2 \\ 0 & 2 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ 3 & 0 \\ 0 & 1 \end{bmatrix}.$$

Окончательно

$$f(X) = \begin{bmatrix} 3 \\ 3 \\ 3 \end{bmatrix}(X') \times \begin{bmatrix} 2 \\ 2 \end{bmatrix}(X'') + \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}(X') \times \begin{bmatrix} 3 \\ 0 \end{bmatrix}(X'') + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}(X') \times \begin{bmatrix} 0 \\ 1 \end{bmatrix}(X''),$$

откуда получим

$$f(x_0, x_1) = 2 + \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}(x_0) \times \begin{bmatrix} 3 \\ 0 \end{bmatrix}(x_1) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}(x_0) \times x_1 = 2 + \tilde{x}_0 \times \bar{x}_1 + \tilde{x}_0 \times x_1,$$

где $\bar{x}$ — унарная операция, преобразующая нуль и единицу переменной x соответственно в единицу и нуль конечного поля.

2.3. Алгебры образующих операций

Существует шесть типов алгебр [25], порождающих различные классы спектральных функций (табл. 5). Все они предназначены для решения сис-

темы линейных алгебраических уравнений (3). Для этого определенные требования налагаются на операции алгебр и вводятся некоторые ограничения на вид спектральных функций.

Алгебра логики имеет нейтральный элемент по сложению σ (нуль) и левый нейтральный элемент по умножению τ (единица). В мультипликативной алгебре операция умножения обращаема на множестве $N_k/\{\sigma\}$. Операция сложения аддитивной алгебры образует коммутативную группу на множестве N_k. В конечном поле сложение и умножение формируют коммутативные группы на своих множествах, а умножение дистрибутивно относительно сложения. В целостном кольце имеется коммутативная группа по сложению, отсутствуют нетривиальные делители нуля и, как и в поле, выполняется дистрибутивный закон. Алгебра Уолша является подкольцом целостного кольца. Ограничения, налагаемые на коэффициенты уравнений d_{ji} в каждой из алгебр, определяются теоремами разложения и задают соответствующий класс спектральных базисов. В частности, в алгебре логики система спектральных функций должна представляться матрицей перестановок, в мультипликативной алгебре — мономиальной матрицей, в аддитивной алгебре — логической матрицей, имеющей ненулевой определитель, по модулю меньший циклического порядка группы по сложению, в конечном поле — обращаемой матрицей, в алгебре Уолша и целостном кольце — матрицей, имеющей ненулевой определитель.

На практике распространение получило небольшое число спектральных функций (табл. 6). В классе унимодальных используются дизъюнктивные [26], конъюнктивные [21], литеральные [27] и интервальные [28] функции. Из многомодальных применяются функции Жегалкина [29], Рида—Малера [30, 31], теоретико-числовые [13, 14] и полиномиальные [15]. Бимодальные функции представлены функциями Радемахера [16], Уолша [32], Хаара [33] и Виленкина—Крестенсона [34, 35].

Таблица 5

Алгебры образующий операций

Функции	Двузначные	Многозначные
Унимодальные	Логики R_L	Мультипликативная R_M
Мультимодальные	Аддитивная R_A	Конечное поле R_F
Бимодальные	Уолша R_W	Целостное кольцо R_R

Таблица 6

Системы спектральных функций

Функции	Двузначные	Многозначные
Унимодальные Мультимодальные Бимодальные	Дизъюнктивная Конъюнктивная Жегалкина Рида—Маллера Радемахера Уолша	Литеральная Интервальная Теоретико-числовые Полиномиальные Хаара Виленкина—Крестенсона

2.4. Функциональная полнота

Как показано выше, при алгебраической декомпозиции используется базис

$$\Theta = \{+, \times, \{\theta_i\}, \{a_j\}\},$$

состоящий из операций образующей алгебры R, фиксированного множества спектральных k'-функций и множества, содержащего все возможные k''-функции, где $k'k'' = m$, а m — длина характеристического вектора декомпозируемой функции. В пределе, когда $k' = m$, имеем спектральную декомпозицию, у которой множество $\{a_j\}$ — константы алгебры R. Базис Ω является функционально полным. Его полнота непосредственно следует из разрешимости системы линейных уравнений (3) благодаря свойствам операций образующей алгебры и ограничениям, накладываемым на спектральные функции.

3. МУЛЬТИПЛИКАТИВНАЯ АЛГЕБРА

3.1. Теоремы разложения

Определение 1. *Алгебраическая система $R_M = \langle N_k, +, \times\rangle$ называется примитивной мультипликативной алгеброй, если существует элемент $\sigma \in N_k$, такой, что для произвольного $a \in N_k$ выполняются равенства $\sigma + a = a$, $a + \sigma = a$, $\sigma \times a = \sigma$ и уравнение $a \times x = b$ имеет единственное решение относительно x при произвольных $a \neq \sigma$ и b.*

Из определения 1 следует, что умножение является операцией (не обязательно ассоциативной), которая имеет левую обратную операцию на множестве $N_k\backslash\{\sigma\}$. Это означает, что матрица операции умножения имеет строку σ, состоящую только из элементов σ, а все остальные строки таковы, что каждый элемент из множества N_k повторяется не более одного раза.

Определение 2. *Мономиальной матрицей называется квадратная матрица, получаемая перестановкой строк (столбцов) диагональной матрицы с элементами $d_i \neq \sigma$.*

Мономиальной будет такая матрица, которая в каждой строке и в каждом столбце содержит только один элемент, отличный от σ. Мономиальная матрица может быть получена из матрицы перестановок заменой единичных элементов на произвольные ненулевые элементы.

Теорема 1. *Произвольная функция f представима в примитивной мультипликативной алгебре R_M в ви-*

де разложения (2), если спектральные функции образуют матрицу прямого преобразования ***D****, которая является мономиальной.*

Доказательство. Из мономиальности матрицы **D** следует, что в каждом уравнении системы (3) только один элемент d_{jt} отличен от σ. Тогда в алгебре R_M система (3) вырождается в систему $f(j, X'') = d_{jt} \times a_t(X'')$ ($j = \overline{0, k'-1}$), откуда могут быть найдены функции $a_t(X'')$ ($t = \overline{0, k'-1}$). Следовательно, система уравнений (3) разрешима в алгебре R_M. Последнее означает, что произвольная функция представима разложением (2).

Пример 5. Пусть операции сложения и умножения примитивной мультипликативной алгебры заданы матрицами

$$+ = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 1 & 1 & 3 & 3 \\ 2 & 3 & 2 & 3 \\ 3 & 3 & 3 & 3 \end{bmatrix}, \quad \times = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 2 & 3 & 0 \\ 2 & 3 & 0 & 1 \\ 3 & 0 & 1 & 2 \end{bmatrix},$$

где сложение — двухразрядная побитовая дизъюнкция, а умножение имеет левую обратную операцию на множестве {1, 2, 3} и представляет собой усеченное сложение по модулю 4. Из матриц операций находим, что σ = 0. Определим систему спектральных функций мономиальной матрицей **D** и для функции из примера 4 запишем систему уравнений $\mathbf{D} \times \mathbf{A} = \mathbf{T}$:

$$\begin{bmatrix} 0 & 3 & 0 \\ 0 & 0 & 1 \\ 2 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} a_0(0) & a_1(0) \\ a_0(1) & a_1(1) \\ a_0(2) & a_1(2) \end{bmatrix} = \begin{bmatrix} 3 & 2 \\ 0 & 2 \\ 1 & 0 \end{bmatrix},$$

откуда, решая каждое уравнение в отдельности, находим матрицу коэффициентов:

$$\begin{cases} 3 \times a_0(1) = 3 \\ 1 \times a_0(2) = 0 \\ 2 \times a_0(0) = 1 \end{cases} \quad \begin{cases} 3 \times a_1(1) = 2 \\ 1 \times a_1(2) = 2 \\ 2 \times a_1(0) = 0 \end{cases} \quad \mathbf{A} = \begin{bmatrix} 3 & 2 \\ 0 & 3 \\ 3 & 1 \end{bmatrix}.$$

Окончательно

$$f(X) = \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}(X') \times \begin{bmatrix} 3 \\ 2 \end{bmatrix}(X'') + \begin{bmatrix} 3 \\ 0 \\ 0 \end{bmatrix}(X') \times \begin{bmatrix} 0 \\ 3 \end{bmatrix}(X'') + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}(X') \times \begin{bmatrix} 3 \\ 1 \end{bmatrix}(X'').$$

При доказательстве теоремы 1 учтены ограничения на операцию умножения, минимально необходимые для разрешимости системы уравнений (3). Однако эти ограничения недостаточны для применения матричного аппарата обратного дискретного преобразования (4). В примитивной алгебре матрица обратного преобразования **Q** зависит от представляемой функции (см. пример 5), в то время как в других образующих алгебрах она определяется только спектральными функциями. Следовательно, матрица обратного преобразования не может быть вычислена заранее, когда представляемая функция еще неизвестна.

Определение 3. *Мультипликативная алгебра называется непримитивной, если операция умножения образует группу* $G_M = \langle N_k \backslash \{\sigma\}, \times \rangle$.

В непримитивной алгебре умножение ассоциативно и уравнения $a \times x = b$ и $x \times a = b$ имеют единственные решения при $a \neq \sigma$. Последнее означает, что существует единственный нейтральный элемент $\tau \neq \sigma$, такой, что $\tau \times a = a \times \tau = a$ и для любого элемента $a \neq \sigma$ существует обратный ему элемент a^{-1} такой, что $a \times a^{-1} = a^{-1} \times a = \tau$.

Теорема 2. *Если матрица* ***D*** *является мономиальной, то произвольная функция f представима в непримитивной мультипликативной алгебре разложением (2), для которого существует матрица обратного преобразования* $\mathbf{Q} = \tilde{\mathbf{D}}^T$*, где матрица* $\tilde{\mathbf{D}}$ *получается из матрицы D заменой ненулевых элементов на обратные.*

Доказательство**.** Умножим левую и правую части матричного выражения (4) на $\tilde{\mathbf{D}}^T$ и получим $\tilde{\mathbf{D}}^T \times \mathbf{T} = \tilde{\mathbf{D}}^T \times \mathbf{D} \times \mathbf{A}$. Так как умножение ассоциативно и $\tilde{\mathbf{D}} \times \mathbf{D} = \mathbf{E}$, где **E** — единичная матрица, то справедливо выражение коэффициентов разложения через матричные операции $\mathbf{A} = \tilde{\mathbf{D}}^T \times \mathbf{T}$, откуда находим $\mathbf{Q} = \tilde{\mathbf{D}}^T$.

Заметим, что в мультипликативной алгебре умножение может быть некоммутативной операцией и отсутствует требование дистрибутивности умножения относительно сложения, как это имеет место в конечном поле и целостном кольце.

Пример 6. Пусть сложение и умножение заданы матрицами

$$+ = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 1 & 1 & 3 & 3 \\ 2 & 3 & 2 & 3 \\ 3 & 3 & 3 & 3 \end{bmatrix}, \quad \times = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 2 & 3 & 1 \\ 0 & 3 & 1 & 2 \\ 0 & 1 & 2 & 3 \end{bmatrix},$$

где сложение, как и ранее, двухразрядная побитовая дизъюнкция, а умножение образует группу на множестве {1, 2, 3}. Из матриц операций находим,

что $\sigma = 0$ и $\tau = 3$. Определим систему спектральных функций мономиальной матрицей **D** и найдем в заданной алгебре матрицу обратного преобразования **Q**:

$$\mathbf{D} = \begin{bmatrix} 0 & 0 & 1 \\ 3 & 0 & 0 \\ 0 & 2 & 0 \end{bmatrix}, \quad \mathbf{Q} = \tilde{\mathbf{D}}^T \begin{bmatrix} 0 & 3 & 0 \\ 0 & 0 & 1 \\ 2 & 0 & 0 \end{bmatrix}.$$

Далее для функции из примера 4 вычислим матрицу коэффициентов:

$$\mathbf{A} = \mathbf{Q} \times \mathbf{T} = \begin{bmatrix} 0 & 3 & 0 \\ 0 & 0 & 1 \\ 2 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} 3 & 2 \\ 0 & 2 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 2 \\ 2 & 0 \\ 2 & 1 \end{bmatrix}$$

Окончательно

$$f(X) = \begin{bmatrix} 0 \\ 3 \\ 0 \end{bmatrix}(X') \times \begin{bmatrix} 0 \\ 2 \end{bmatrix}(X'') + \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}(X') \times \begin{bmatrix} 2 \\ 0 \end{bmatrix}(X'') + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}(X') \times \begin{bmatrix} 2 \\ 1 \end{bmatrix}(X'').$$

Нетрудно показать, что число представлений произвольной функции в мультипликативной алгебре равно $k'!(k-1)^{k'}$, где k — значность мультипликативной алгебры, k' — длина характеристического вектора спектральных функций. При спектральном разложении это число максимально и равно $m!(k-1)^m$, где m — длина характеристического вектора декомпозируемой функции.

3.2. Мультипликативные формы

В мультипликативной алгебре спектральные функции образуют мономиальную матрицу, т. е. принимают ненулевые значения только в одной точке области определения, причем различные функции в различных точках. По классификации из табл. 5 находим, что эти функции относятся к классу многозначных унимодальных.

Поставим задачу оптимального синтеза дискретной функции в мультипликативной алгебре. Оптимальной будем называть такую форму, у которой число операций (логических элементов), требуемое для вычисления (реализации) функции минимально.

Вначале поставим задачу так определить алгебру и систему спектральных функций, чтобы количество слагаемых m' в выражении (3) было минимальным, $m' \le k'$. Заметим, что некоторое слагаемое будет отсутствовать в разложении, если единственное ненулевое значение спектральной функции $\theta_i(X')$, будучи умноженное на функцию-коэффициент $a_i(X'')$ дает нуль алгебры σ. Так как уравнение $d \times a = \sigma$ имеет единственное решение относительно a при $d \ne \sigma$, то функция-коэффициент $a_i(X'')$ должна не зависеть от переменных из множества X'', т. е. равняться некоторой константе. Следовательно, в качестве нуля алгебры следует взять элемент, из которого состоит наибольшее число константных строк матрицы функции. При спектральном разложении, когда все коэффициенты константы, нуль алгебры должен совпадать с наиболее часто встречающимся значением в характеристическом векторе. С другой стороны, для каждой функции-коэффициента можно задать свою спектральную функцию. Последнее означает, что путем выбора спектральных функций число умножений в разложении может быть уменьшено на количество константных строк, имеющихся в матрице функции, а наиболее часто повторяющиеся элементы в характеристических векторах коэффициентов преобразованы в один и тот же элемент.

3.3. Демонстрационный пример

Рассмотрим функцию из примера 3. Для ее представления в мультипликативной форме будем пользоваться ранее определенной в примере 5 примитивной алгеброй.

Для начала зададим систему спектральных функций произвольной диагональной матрицей и вычислим матрицу коэффициентов. Представим результаты вычислений в виде системы уравнений вида $\mathbf{T} = \mathbf{D} \times \mathbf{A}$:

$$\begin{bmatrix} 2 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 3 & 1 & 3 & 1 \\ 2 & 2 & 2 & 2 \\ 1 & 3 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 \end{bmatrix} \times \begin{bmatrix} 3 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 \\ 0 & 3 & 0 & 3 \\ 3 & 3 & 3 & 3 \\ 2 & 0 & 1 & 2 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

Теперь определим спектральные функции так, чтобы наибольшее число коэффициентов стало равным некоторому элементу $\delta \in N_4$. Для этого потребуем, чтобы уравнение $x \times \delta = b$ имело единственное решение относительно x для произвольных b. В примитивной алгебре, определенной в примере 5, это возможно только при $\delta = 0$, в непримитивной алгебре, когда умножение образует группу, δ может быть любым, кроме нуля алгебры.

Представим теперь заданную функцию в непримитивной алгебре из примера 6. При заданном разделении переменных находим, что в нулевой строке наибольшее число раз встречается 1, в первой строке — 0, во второй — 3 и 1, в третьей — 2, в четвертой — 1 и в пятой 0. Для уменьшения ко-

личества операций умножения положим $\delta = \tau$, так как известно, что $a \times \tau = a$. Поскольку уравнение $x \times \tau = b$ имеет единственное решение для произвольных b, находим

$$\begin{cases} d_0 \times 3 = 1 \\ d_1 \times 3 = 0 \\ d_2 \times 3 = 3 \\ d_3 \times 3 = 2 \\ d_4 \times 3 = 1 \\ d_5 \times 3 = 0 \end{cases}, \quad \begin{cases} d_0 = 1 \\ d_1 = 0 \\ d_2 = 3 \\ d_3 = 2 \\ d_4 = 1 \\ d_5 = 0 \end{cases}, \quad \Delta = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Нулевые столбцы в матрице **D** не нарушают условия теоремы разложения, а только лишь указывают на то, что данная спектральная функция не используется в разложении. Так как имеется возможность приравнять функцию-коэффициент к нулю, то нулевая спектральная функция может быть произвольной (показано знаком *),

$$\begin{bmatrix} 2 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 3 & 1 & 3 & 1 \\ 2 & 2 & 2 & 2 \\ 1 & 3 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & * & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & * \end{bmatrix} \times \begin{bmatrix} 1 & 3 & 3 & 3 \\ 0 & 0 & 0 & 0 \\ 3 & 1 & 3 & 1 \\ 3 & 3 & 3 & 3 \\ 3 & 2 & 0 & 3 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Переставим столбцы матрицы **D** так, чтобы тривиальные функции-коэффициенты были последними в разложении:

$$\begin{bmatrix} 2 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 3 & 1 & 3 & 1 \\ 2 & 2 & 2 & 2 \\ 1 & 3 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & * & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & * \end{bmatrix} \times \begin{bmatrix} 3 & 2 & 0 & 3 \\ 3 & 1 & 3 & 1 \\ 1 & 3 & 3 & 3 \\ 3 & 3 & 3 & 3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

В итоге

$$f(X) = \theta_0(X') \times \begin{bmatrix} 3 \\ 2 \\ 0 \\ 3 \end{bmatrix}(X'') + \theta_1(X') \times \begin{bmatrix} 3 \\ 1 \\ 3 \\ 1 \end{bmatrix}(X'') +$$

$$+ \theta_2(X') \times \begin{bmatrix} 1 \\ 3 \\ 3 \\ 3 \end{bmatrix}(X'') + \theta_3(X').$$

Для получения окончательного выражения осталось представить функции-коэффициенты в виде выражений, возможно, в другой алгебре.

4. РЕАЛИЗАЦИЯ МУЛЬТИПЛИКАТИВНЫХ ФОРМ

Поставим задачу поиска аналитических конструкций, позволяющих эффективным образом представлять многозначные унимодальные функции в виде выражений в некотором базисе операций.

Так как ненулевое значение многозначная унимодальная функция принимает только в одной точке, то такие функции могут быть отнесены к первичным, т. е. не нуждающимся в разложении. Табличное вычисление этих функций представляется малоэффективным, так как их характеристические вектора состоят, в основном, из одного и того же элемента и этот элемент равен нулю мультипликативной алгебры. Следовательно, необходимо найти операции, позволяющие эффективно вычислять многозначную унимодальную функцию, и поиск таких операций следует осуществлять с учетом операционных возможностей применяемых вычислительных средств (при программной реализации) или применяемой элементной базы (при аппаратной реализации).

При программной реализации многозначные унимодальные функции легко вычисляются с помощью операций сравнения и условного перехода. Реализация алгебраических операций также не представляет трудности. Так как необходимо обеспечить возможность выбора нуля алгебры, сложение следует реализовать с помощью поразрядной операции суммы по модулю 2 следующим образом: $a + b = a \oplus b \oplus \sigma$. Операция умножения примитивной алгебры может быть построена на основе арифметического сложения путем его усечения, т. е. установки результата умножения в σ, когда первый операнд равен σ. При этом, очевидно, необходимо использовать операцию сравнения, сложения и условного перехода. Умножение непримитивной алгебры также может быть построено на основе операции сложения путем ее усечения по двум операндам и смещения результата сложения на 1.

Аппаратная реализация мультипликативных форм также не представляет трудности и осуществляется аналогично программной. Для этого применяются такие элементы, как логические компараторы и конъюнкторы — для реализации унимодальных функций, дизъюнкторы или сумматоры по модулю 2 — для объединения результатов вычисления спектральных функций и сумматоры — для реализации операции умножения.

ЗАКЛЮЧЕНИЕ

Применение мультипликативной алгебры для алгебраической декомпозиции дискретной функции позволяет по виду функции, до проведения синтеза, определить эффективность ее формульного представления.

Другое важное свойство мультипликативной формы состоит в возможности управлять значениями коэффициентов. В частности, при спектральном разложении может быть синтезирована форма, все коэффициенты которой нули или единицы, т. е. результирующее выражение не будет содержать операций умножения.

Путем перестановки (задания другой нумерации) многозначных унимодальных функций возможно получение форм, у которых первоначально вычисляются наиболее вероятные функции и, в случае неравенства результата вычисления нулю, дальнейшие вычисления прекращаются, а вычисленное значение однозначно определяет итоговый результат.

Перечисленные свойства отличают мультипликативную форму от известных форм и позволяют свести синтез минимальных форм дискретных функций к поиску такого разделения переменных, при котором ее прямоугольная таблица содержит наибольшее число константных строк (в противоположность разделительной декомпозиции, когда ищется наибольшее число одинаковых строк).

ЛИТЕРАТУРА

1. *Яблонский С. В.* Об алгоритмических трудностях синтеза минимальных контактных схем // Проблемы кибернетики. Вып. 2. М.: Физматгиз, 1959. – С. 75—121.
2. *Coloring* Large Graph // Proc. of the 1981 S.E. Conf. on Graph Theory, Combinatorics and Computer Science. — 1981.
3. *Малюгин В. Д.* Реализация булевых функций арифметическими полиномами // Автоматика и телемеханика. — 1982. — № 4. — С. 84—93.
4. *Малюгин В. Д.* Реализация кортежей булевых функций посредством линейных арифметических полиномов // Автоматика и телемеханика. — 1984. — № 2. — С. 114—122.
5. *Выхованец В. С., Малюгин В. Д.* Кратные логические вычисления // Автоматика и телемеханика. — 1998. — № 6. — С. 163—171.
6. *Lee C. Y.* Representation of switching circuits by binary decision programs // Bell System Technical Journal. — 1959. — Vol. 38, No. 4. — P. 985.
7. *Akers S. B.* Binary decision diagrams // IEEE Trans. Computers. — 1978. — Vol. C-27, No. 6. — P. 509—516.
8. *Bernstein B.* Operations with Respect to wich the Elements of Boolean Algebra from a Group // Trans. Amer. Math. Soc. — 1924. — Vol. 26. — P. 171—175.
9. *Cohn M.* Switching Functions Canonical Form over Integer Fields (Ph.D. Thesis). — Cambridge: Harvard Univ., 1960.
10. *Мерекин Ю. В.* Арифметические формы записи булевых выражений и их применение для расчета надежности схем // Вычислительные системы. Вып. 7. — Новосибирск: Ин-т математики СО АН СССР, 1963. — С. 13—23.
11. *Tosic Z.* Analytical Representation of an m–Valued Logical Function over the Ring of Integers Modulo m (Ph.D. Thesis). — Beograd, 1972.
12. *Green D. H.* Reed-Muller Expansions of Incompletely Specified Functions // Proc. IEE. — 1987. — Part E-134. — P. 228—236.
13. *Фараджиев Р. Г., Ципкин Я. З.* Преобразования Лапласа–Галуа в теории последовательных машин // Докл. АН СССР. — 1966. — Т. 166, № 36.
14. *Rader C. M.* Discrete convolution via Mersenne transorm // IEEE Trans. Comp. — 1972. — Vol. C-21.
15. *Макклеллан Дж. Х., Рейдер Ч. М.* Применение теории чисел в цифровой обработке сигналов. — М.: Радио и связь, 1983.
16. *Rademacher H.* Einige Satze uber Reihen von allgemeinen Orthogonalfunktionen // Math. Ann. — 1922. — Vol. 87. P. 112—138.
17. *Закревский А. Д.* Алгоритм декомпозиции булевых функций // Тр. Сибирского физико-техн. ин-та. — 1964. — Вып. 44. — С. 5—16.
18. *Ashenhurst R. L.* The Decomposition of Switching Functions // Bell Laboratory Report. — 1952. — No. BL-1(11). — P. 541—642.
19. *Поваров Г. Н.* О функциональной разделимости булевых функций // Докл. АН СССР. — 1954. — Т. 94. — С. 801—803.
20. *Curtis H. A.* Non-Disjunctive Decomposition // Bell Laboratory Report. — 1958. — No. 19. — P. 49.
21. *Shannon C.E.* The Synthesis of Two-Terminal Switching Circuits // Bell System Technical Journal. — 1949. — No. 28. — P. 59—98. — Vol. 29. – P. 74—116.
22. *Lechner R. J.* Harmonic Analysis of Switching Functions // Recent Developments in Switching Theory. — Academic Press, 1971. — P. 121—228.
23. *Карповский М. Г., Москалев Э. С.* Спектральные методы анализа и синтеза дискретных устройств. — М.: Энергия, 1973.
24. *Лупанов О. Б.* Об одном методе синтеза схем // Изв. вузов. Радиофизика. — 1958. — № 1. — С. 120—140.
25. *Выхованец В. С.* Спектральные методы в логической обработке данных // Автоматика и телемеханика. — 2001. — № 10. — С. 28—53.
26. *Boole G.* The Laws of Thought. — London: Macmillan, 1854.
27. *Яблонский С. В.* Функциональные построения в *k*-значной логике // Тр. Матем. ин-та АН СССР им. В. А. Стеклова. — 1958. — Т. 51. — С. 5—142.
28. *Vranesic Z. C., Lee E. S., Smith K. S.* A Many-Valued Algebra for Switching Systems // IEEE Trans. Computers. — 1970. — Vol. C-19. — P. 964—971.
29. *Жегалкин И. И.* О технике вычисления предложений в символической логике // Математический сборник. — 1927. — Т. 43. — С. 9—28.
30. *Reed L. S.* A class of multiple error correction codes and their decoding scheme // IRE Trans. on Inform. Theory. — 1954. — Vol. 4. — P. 38—42.
31. *Muller D. E.* Application of Boolean algebra to switching circuit design and to error detection // IRE Trans. Electron. Comput. — 1954. — Vol. EC-3. — P. 6—12.
32. *Walsh J. L.* A closed set of orthogonal functions // Amer. J. Math. — 1923. — Vol. 55. — P. 5—24.
33. *Haar A.* Zur Theorie der orthogonolen Funktionensysteme // Math. Ann. — 1910. — Vol. 69. — P. 331—371.
34. *Виленкин Н. Я.* Класс полностью ортогональных систем // Изв. АН СССР. Сер. Математика. — 1947. – № 11. — С. 363—400.
35. *Chrestenson H. E.* A class of generalized Walsh functions // Pacific J. Math. — 1955. — Vol. 5. — P. 17—31.

☎ *(095) 787-88-44*

E-mail: vyk@ipu.ru

❑

УДК 681.324

РАБОТА ЛОКАЛЬНЫХ КОМПЛЕКСОВ GRID В РЕЖИМЕ ЖЕСТКОГО РЕАЛЬНОГО ВРЕМЕНИ

Г. Г. Стецюра

Институт проблем управления им. В. А. Трапезникова, г. Москва

Рассмотрено применение в локальных GRID групповых операций и совмещения процессов передачи и обработки данных, позволяющее на предлагаемых GRID решать задачи жесткого реального времени.

ВВЕДЕНИЕ

Существует два типа сетей GRID — глобальные, с выходом в сети дальней связи, например, в Internet, и локальные, в которых компьютеры объединены локальной сетью. Сети GRID обычно рассматриваются как распределенный компьютер, однако характер выполнения вычислений в GRID весьма специфичен. Основная задача объединенных сетью компьютеров — обслуживать владельцев этих компьютеров, а распределенная по сети задача решается в фоновом режиме. Алгоритмы решения распределенных задач в глобальных GRID создаются с учетом непредсказуемой задержки вычислений и частого отключения удаленных компьютеров. Локальные сети GRID могут предоставлять сервис, существенно отличающийся от того, который дают глобальные GRID. Тем не менее, принципиальных отличий в реализованных сетях нет. Например, широко известная локальная GRID Condor [1] функционирует подобно глобальной GRID, но является более скоростной.

В настоящей статье рассмотрено использование локальных сетей GRID для решения задач жесткого реального времени (ЖРВ). Это могут быть задачи управления работой технических средств или вычислительные задачи, в которых требуется выдерживать временную согласованность выполнения частей задачи на разных компьютерах сети.

Будем называть GRID, способную решать такие задачи, комплексом GRID, чтобы подчеркнуть наличие в ней дополнительных структурных и технических решений, приближающих сетевую структуру к компьютеру.

Существуют два режима, в которых GRID может решать задачи ЖРВ. Во-первых, после окончания рабочего дня пользователи покидают свои компьютеры, и для всей совокупности таких компьютеров распределенная задача становится основной. Во-вторых, даже при наличии пользователей, для распределенной задачи, требующей режима ЖРВ, можно, как это обычно делается в сетях для передачи срочных сообщений, выделить квант времени, который гарантированно предоставляется распределенной задаче. Во втором варианте права владельцев ущемляются, но появляется возможность гарантированно управлять ответственными процессами, например, физическим экспериментом. Таким образом, здесь речь идет о расширении функций локальных сетей GRID.

Легко прослеживается аналогия между указанными режимами работы GRID и работой автоматизированных систем управления, в частности, промышленных АСУ. Для таких систем в Институте проблем управления РАН ранее были разработаны методы, позволяющие существенно повысить реактивность систем [2, 3]. В последние годы они были дополнительно развиты и предлагаются в этой статье для применения в локальных комплексах GRID.

Центральная особенность разработанных методов состоит в использовании групповых операций. Под групповой операцией здесь понимается логическая или арифметическая операция, которую выполняет группа взаимодействующих устройств над данными, хранящимися в этих устройствах. При этом вычисления производятся в процессе передачи промежуточных результатов операции по каналу без их задержки на обработку [2]. Указанная независимость времени выполнения операции от количества участвующих в ней компьютеров — основная особенность групповых операций. Отсутствие задержки на вычисление достигается за счет совме-

щения обработки данных с их передачей по каналам связи.

Ниже кратко изложены особенности разработанных в ИПУ методов и применение их для реализации функций GRID в указанных выше режимах работы.

1. ПРИОРИТЕТНОЕ УПРАВЛЕНИЕ ДОСТУПОМ К РЕСУРСАМ

В работах [2, 4] изложены способы децентрализованного приоритетного управления (ДПУ) доступом компьютеров к средствам связи. Управление осуществляется на основе приоритетов компьютеров. Приоритеты могут изменяться динамически с учетом потребностей компьютеров и обеспечивают однозначный выбор компьютера, которому и предоставляется доступ. Способы ДПУ, прежде всего, обеспечивают доступ за гарантированное время, что необходимо для систем ЖРВ. Однако обеспечивается не только доступ к средствам связи. Средства связи — лишь общая среда общения компьютеров. Приоритеты ДПУ могут относиться к любым ресурсам GRID. Запросы компьютеров могут вначале отбираться, исходя из доступности ресурсов, которые требуются компьютерам. Только после этого, для обеспечения однозначности доступа, привлекаются другие параметры. Время доступа при ДПУ логарифмически зависит от числа компьютеров.

В способах ДПУ используются сигналы, которые позволяют обнаружить наличие в канале связи сигналов, соответствующих двоичной единице, при одновременном с ними нахождении сигналов двоичного нуля. Этого достаточно для выполнения операции ***max***, которая последовательно сравнивает разряды передаваемых двоичных кодов приоритетов и выделяет наибольшее значение приоритета.

В ИПУ были разработаны также способы доступа, решающие рассмотренную задачу на основе введения различных временных задержек сигнала, соответствующих разным приоритетам — это методы децентрализованного пространственно-временного управления.

2. СОВМЕЩЕНИЕ ОБРАБОТКИ ДАННЫХ С ИХ ПЕРЕДАЧЕЙ

Для проведения обработки данных без задержки (“на лету”), в процессе передачи используется канал с направленной передачей данных, в котором от компьютера к компьютеру последовательно передаются сигналы. Сигнал по каналу поступает к одному из компьютеров, без задержки обрабатывается им, преобразованный сигнал поступает к

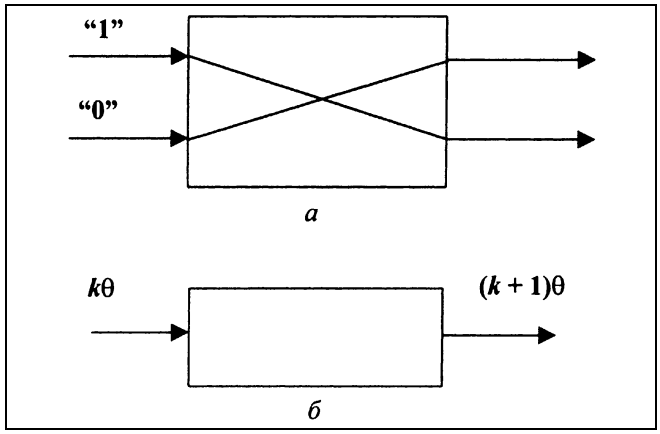


Рис. 1. Варианты инвертирования сигнала:
а — переключение линий; *б* — вращение плоскости поляризации

следующему компьютеру и т. д. Без задержки можно выполнить операцию только в том случае, если для выбора необходимого действия не требуется знать значение приходящего по каналу сигнала. Трудности при этом возникают с единственной операцией — инвертированием значения сигнала. Так как предполагается использование двоичных сигналов, то речь идет о превращении сигнала “1” в сигнал “0” и наоборот. На рис. 1 показаны способы выполнения инвертирования с нулевой временной задержкой [4]. На рис. 1, *а* сигнал в инвертор поступает по одной из двух линий и выходит из него также по двум линиям. Если сигнал поступает по верхней линии, то это соответствует приходу двоичной единицы. Сигнал на нижней линии соответствует двоичному нулю. В инверторе линии перекрещены, что и обеспечивает инвертирование значения сигнала. На рис. 1, *б* показан другой вариант инвертирования [5]. Сигнал предполагается поляризованным. Он проходит через среду, вращающую плоскость поляризации сигнала на угол $\theta = 360^\circ/n$, где n — четное число. При выполнении этого условия по поляризации сигнала можно определить его двоичное значение, во-вторых, инвертирование, как и в первом варианте реализации этой операции, происходит без задержки.

Отметим отличие приведенных элементов от других известных реализаций. В последних входной сигнал не покидает элемент. Его энергия используется для управления созданием выходного сигнала за счет внешнего источника энергии. В разработанных в ИПУ вариантах входной сигнал передается непосредственно на выход элемента.

С помощью указанных преобразований реализуются логические и арифметические операции. В публикациях [2, 4] такие операции названы “вы-

Рис. 2. Структура групповой команды

численим в общем канале" (ВОК). Несмотря на последовательную поразрядную передачу операнда ВОК выполняется очень быстро — за время перемещения операнда по каналу выполняется количество операций, равное количеству компьютеров комплекса. Например, для суммирования N чисел, распределенных по одному между N компьютерами, по каналу передается значение суммы, которое последовательно формирует каждый из N компьютеров. Это оказывается возможным только вследствие отсутствия задержки при выполнении ВОК. В работе [6] приведены области вычислений, где применение ВОК наиболее эффективно.

3. ГРУППОВЫЕ КОМАНДЫ, РАСПРЕДЕЛЕННЫЕ ПРОГРАММЫ

Групповые команды (ГК) — основа рассматриваемого в этом разделе распределенного программирования [5]. Их можно представить как сообщение, состоящее из трех блоков, показанных на рис. 2. Первый блок содержит код операции, требующий от компьютеров комплекса определенного действия. Второй блок содержит набор операндов. Компьютеры используют этот блок совместно с хранящимися в них данными для получения результата операции. Третий блок — блок результатов — содержит результаты выполнения команды. Он хранит результаты двух видов. Это может быть последовательность индивидуальных результатов, поочередно переданных в ГК участвовавшими в ее выполнении компьютерами. Но блок может также содержать результаты совместной работы компьютеров, выполненные с применением ВОК. Пример — создание суммы чисел непосредственно в групповой команде. Таким образом, ГК, перемещаясь между компьютерами, наполняется результатами заданной в ней операции, и за один обход всех компьютеров выполняется сложная совокупность действий.

В распределенной программе, естественно, необходимы операции ветвления процесса вычисления. Помимо традиционных способов ветвления здесь возможны ветвления, требующие одновременного участия большого количества компьютеров. При таком ветвлении источник программы посылает всем компьютерам групповую команду, указывающую, какими свойствами должны обладать компьютеры, чтобы иметь право в дальнейшем получить управление обработкой данных. Из этих компьютеров должен быть выделен один. В § 1 показано, как это делается.

Таким образом, при распределенном программировании последовательно выполняются групповые команды.

4. СТРУКТУРЫ СВЯЗЕЙ В ЛОКАЛЬНЫХ GRID С ГРУППОВЫМИ ОПЕРАЦИЯМИ

В рассматриваемых комплексах GRID применимы многие виды структур связи [2, 4]. Здесь остановимся на новой, фрактальной структуре канала связи, разработанной специально для комплексов, использующих групповые операции [4]. Канал связи имеет вид фрактальной кривой Серпинского и строится итерациями с применением L-системы Линденмауэра [7]. На рис. 3 показан канал, полученный после второй итерации. Структура связи имеет регулярный вид, ребра графа на рис. 3 представляют собой каналы связи, вершины — компьютеры с дополнительным оборудованием для выполнения ГК. Вся структура разделяется на отдельные кластеры с целью разделения потоков данных в комплексе и изоляции его дефектных частей [4]. На рис. 3 пунктиром показаны места, в которых целесообразно производить указанные разделения.

В реальных GRID каналы связи на рис. 3 могут иметь различную длину, как это показано на рис. 4.

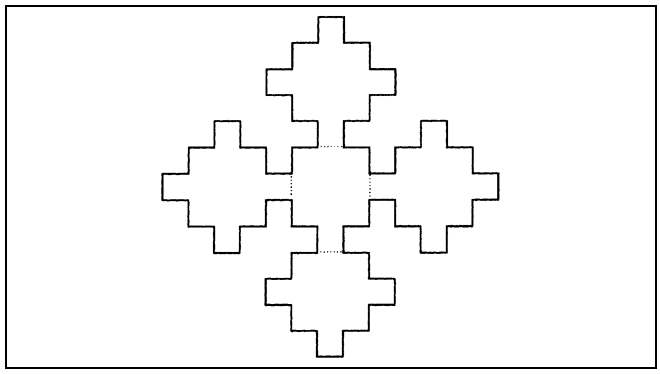


Рис. 3. Фрактальный канал

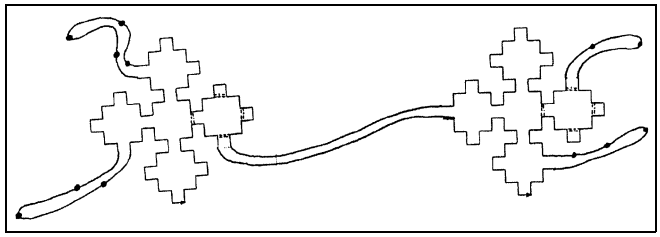


Рис. 4. Фрактальные связи с серверами

Здесь части со связями равной длины будем считать расположенными в сервере комплекса. Удаленные связи подходят непосредственно к компьютерам на рабочих местах (показаны точками). Рассмотренные выше разбиения на кластеры производятся в пределах сервера. На рис. 4 показаны два сервера, каждый их них — это многопроцессорная структура с указанными фрактальными связями. В этой структуре компоненты серверов и клиентские компьютеры включаются в единый процесс обработки информации с применением групповых операций.

5. ГРУППОВЫЕ КОМАНДЫ И НАДЕЖНЫЕ ШИРОКОВЕЩАТЕЛЬНЫЕ СООБЩЕНИЯ

Надежность широковещательных сообщений совершенно необходима для режима ЖРВ. Должна быть гарантия того, что все участвующие в операции компьютеры приняли сообщение. Если это условие не соблюдается, то для корректировки действий надо как можно быстрее выяснить подробности сложившейся ситуации. Групповые команды здесь помогают следующим образом. Групповая команда проверяет работоспособность приемников широковещательного сообщения. В блоке результата этой команды приемники с помощью ВОК формируют сумму, к которой каждый работоспособный приемник прибавляет по единице. Получив эти данные, источник команды определяет состояние комплекса в целом. Если это не случайный сбой, то производится реорганизация системы. Посылается еще одна групповая команда, которая соберет в своем блоке результатов сведения о неработоспособных приемниках [4]. Далее, как показано выше в примере с ветвлением распределенной программы, источник запрашивает замену, и к решению задачи подключаются другие компьютеры. Аналогичным способом могут периодически осуществляться распределенная диагностика комплекса и восстановление его работоспособности.

Групповые команды позволяют быстро определять не только поступление данных в компьютеры, но и их идентичность. Самый быстрый способ состоит в следующем. Каждый компьютер вычисляет контрольную сумму принятого данного. После этого посылается групповая команда, в блоке результата которой компьютеры складывают контрольные суммы. Так как источник знает контрольную сумму посланного им сообщения и количество приемников, то он способен определить идентичность всех принятых сообщений. Так обнаруживаются и сообщения с правильной контрольной суммой, но не соответствующие посланному широковещательному сообщению.

Рассмотренные в настоящем разделе операции были применены в АСУТП [3].

6. СИНХРОНИЗАЦИЯ РАСПРЕДЕЛЕННЫХ РАБОТ В ЛОКАЛЬНОЙ GRID

Далее приведены два характерных для GRID примера синхронизации распределенных работ.

6.1. Распределенная общая память

Одна из важных задач режима ЖРВ заключается в синхронизации блоков распределенной общей памяти (РОП) — частей общей памяти комплекса, расположенных в разных компьютерах [8]. Каждый компьютер решает только часть общей задачи, но для ее решения требуется иметь полную информацию о состоянии объекта управления, создаваемую в разных местах комплекса. Для этого источник должен рассылать новые данные в удаленные блоки РОП. При этом требуется, во-первых, разрешить только поочередную работу источников данных, во-вторых, передающий данные источник должен быть уверен, что процедура обновления содержимого РОП прошла успешно. Быстрое и надежное копирование данных осуществимо с применением методов, изложенных в § 5. Накладные расходы на проверку завершения операции здесь минимальны.

6.2. Синхронизация выполнения идентичных заданий

В GRID поведение распределенных компьютеров непредсказуемо — задание на вычисление может прерываться другим, более приоритетным заданием, компьютер может быть выключен пользователем и т. п. Поэтому на практике одно и то же задание посылается на выполнение группе компьютеров, их результаты заказчик собирает и формирует из них окончательный результат.

Групповые операции существенно ускоряют управление такой процедурой. Выделение группы для выполнения задания выполняется способом, аналогичным поиску компьютеров при передаче управления в распределенной программе (см. § 3). Но здесь выделяется не один, а несколько компьютеров. Количество выделенных компьютеров определяется посылкой одной групповой операции. Так же быстро выполняется передача исходных данных и сбор результатов. Для наблюдения за продвижением процесса вычисления достаточно послать ГК управления с несколькими полями результата в блоке результата. Задается насколько уровней оценки продвижения вычисления и каждому уровню ставится в соответствие один результат. Каждый компьютер группы прибавляет единицу к тому результату, который соответствует уровню продвижения вычислений в данном ком-

пьютере. Фактически в блоке результата ГК строится гистограмма. Общение с компьютерами производится с указанием имени результата, к которому отнес себя компьютер.

7. О ПРОГРАММНОЙ И АППАРАТНОЙ РЕАЛИЗАЦИИ КОМПЛЕКСОВ GRID

Необходимые для построения комплексов GRID аппаратура и программы достаточно просты. Сложность состоит в отличии протокола взаимодействия компьютеров в комплексе GRID от обычно применяемого в локальных сетях протокола TCP/IP. Заметим, что практика применения в специализированных локальных сетях протоколов, отличных от TCP/IP, довольно распространена. Так, в промышленных сетях широко применяется протокол I^2C, многие микроконтроллеры имеют встроенные средства, поддерживающие этот протокол. Расширение приложений сетей, например, создание сетей интеллектуальных датчиков, также порождает новые, существенно отличные от традиционных протоколы [9].

Однако для локальных сетей общего применения, а именно на такие должны ориентироваться комплексы GRID, практически обязательно использование TCP/IP. Поэтому надо уметь совмещать в одной системе два протокола — TCP/IP для локальной сети и протокол группового взаимодействия для объединения компьютеров комплекса. Для этого проще всего потоки данных, относящиеся к разным протоколам, посылать по разным каналам. С этой целью, например, в локальных сетях Ethernet достаточно задействовать имеющиеся в кабелях связи таких сетей неиспользуемые линии.

ЗАКЛЮЧЕНИЕ

Применение коллективного взаимодействия компьютеров при выполнении групповых операций — важный шаг в превращении локальной GRID в действительно распределенный компьютер. Еще раз подчеркнем различие между организацией обработки данных в GRID без групповых операций и с ними. В первом случае стремятся распределить между компьютерами предельно независимые части задачи. Взаимодействие с ними осуществляется обычным сетевым способом. Во втором случае на совокупности компьютеров решается общая задача, и для выполнения вычислений используются средства связи.

ЛИТЕРАТУРА

1. *Thain D., Tannenbaum T., Livny M.* Condor and the Grid // Grid Computing: Making The Global Infrastructure a Reality. — John Wiley, 2003.
2. *Прангишвили И.В., Подлазов В.С., Стецюра Г.Г.* Локальные микропроцессорные вычислительные сети. — М.: Наука, 1984.
3. *Алленов А. В., Подлазов В. С., Слободчук, Смирнов А. Н., Стецюра Г. Г.* Локальные управляющие сети для тесно связанных распределенных систем жесткого реального времени // Вычислительная техника, системы управления. — 1991. — Вып. 6.
4. *Стецюра Г. Г.* Возможности применения фрактальных связей и групповых операций в многопроцессорных системах с перестраиваемой структурой для эволюционных вычислений// Автоматика и телемеханика. — 2003. — № 12.
5. *Стецюра Г. Г.* Возможности технической реализации клеточных автоматов с дальними связями и групповыми операциями // Автоматика и телемеханика. — 2004. — № 8.
6. *Подлазов В. С., Смирнов А. Н., Стецюра Г. Г.* Вычисления в общем канале // Многопроцессорные вычислительные системы с перестраиваемой структурой — М.: Научный совет по комплексной проблеме "Кибернетика", 1981.
7. *Кроновер Р. М.* Фракталы и хаос в динамических системах. — М.: Постмаркет, 2000.
8. *Илларионов Ю. А., Стецюра Г. Г.* Взаимодействие процессов при разделении ресурсов в распределенных вычислительных системах // Измерения, контроль, автоматизация. — 1987. — № 3.
9. *Стецюра Г. Г.* Групповые команды для сетей мобильных датчиков// Датчики и системы. — 2004. — № 4.

☎ *(095) 334-78-31*

E-mail: stetsura@ipu.ru ❑

ВНИМАНИЕ!

Журнал "Проблемы управления" на компакт-диске

Появилась возможность приобретения компакт-диска, полностью воспроизводящего все номера журнала "Проблемы управления" за 2003 г.

ЭТО УДОБНО И НЕДОРОГО

Стоимость диска равна примерно половине стоимости одного номера журнала. Его смогут приобретать не только библиотеки, но и кафедры вузов, отделы и лаборатории, ученые и специалисты.

Цена диска — договорная.

Заказать диск можно в редакции журнала "Проблемы управления": 117997, ГСП-7, Москва, Профсоюзная ул., 65, офис 104

Тел./факс (095) 330-42-66, тел. 334-92-00

E-mail: datchik@ipu.ru

СТАНИСЛАВ ВАСИЛЬЕВИЧ ЕМЕЛЬЯНОВ
(к 75-летию со дня рождения)

Станислав Васильевич Емельянов родился 18 мая 1929 года в Воронеже. Мать — учительница, отец — военнослужащий. Станиславу Васильевичу выпало суровое детство: он рано лишился отца, тяжело заболел и многие годы провел вдали от семьи в лечебницах.

Трудное начало закалило характер, приучило к самостоятельности, воспитало волю. Лишенный многих детских радостей, он много читает, увлеченно учится. В 12 лет он возвращается в семью, вскоре начинается война. После окончания средней школы, движимый желанием учиться, С. В. Емельянов едет в Москву и поступает в Московский авиационный институт на факультет летательных аппаратов. Интерес к авиации он сохранил на всю жизнь.

С. В. Емельянова зачислили в МАИ с условием, что он укрепит здоровье, ибо при росте в 185 см он весил всего 54 кг. Ему рекомендовали заняться плаванием. С трудом (в то время в Москве было всего два бассейна) он выхлопотал разрешение плавать за воротами ватерпольной команды. Со временем он стал играть за команду ватерполистов, выступал на первенстве страны и едва не стал мастером спорта.

Пытливый и хорошо подготовленный юноша привлек внимание академика Б. Н. Петрова — одного из крупнейших ученых в области теории автоматического регулирования, и он пригласил С. В. Емельянова сначала на кафедру МАИ “Системы автоматического управления летательными аппаратами” для выполнения дипломной работы, а затем в 1952 г. и в аспирантуру Института автоматики и телемеханики (ИАТ) — ведущего института по проблемам управления. Уже в кандидатской диссертации (1957 г.) С. В. Емельянов продемонстрировал незаурядный творческий потенциал. Нетрадиционным способом им была решена одна из центральных задач автоматики — управление объектом с сильной изменчивостью параметров в широких пределах. Впервые им было показано, что, комбинируя линейные обратные связи, можно добиться в замкнутой системе гораздо большего, нежели при фиксированной линейной обратной связи. Впервые в мировой практике начато регулярное использование неустойчивых движений и неустойчивых структур для улучшения качества систем автоматического управления. Перспективы развития новых идей в теоретическом и прикладном аспектах выглядели настолько привлекательными, что руководство ИАТ создало лабораторию, задача которой состояла в интенсивном развитии нового направления. Заведующим лабораторией стал канд. техн. наук С. В. Емельянов.

Вокруг него сплотились молодые энтузиасты и за относительно короткий срок были разработаны основы теории систем автоматического управления с переменной структурой (СПС), составившие основу докторской диссертации С. В. Емельянова (1964 г.), а затем — и относительно полная теория СПС, изложенная в монографии коллектива авторов (1970 г.). В теории СПС акцент сделан на использование скользящих режимов, именно в этом случае достигается полная независимость уравнений движения от факторов неопределенности (возмущений параметров и внешних сил). В СПС, работающих в скользящем режиме, выявился ряд дополнительных факторов, повлиявших на популярность этого подхода. Прежде всего, это понижение в скользящем режиме порядка уравнений движения замкнутой системы, что значительно упрощает ее анализ. Кроме того, задача синтеза естественным образом распадается на совокупность двух задач меньшей размерности. Более того, СПС оказались грубыми по отношению к различного рода неидеальностям, в частности, к запаздыванию в переключающем устройстве, к немодели-

руемой динамике и т. д. В теории СПС весьма эффективно решались многие актуальные задачи теории управления, в том числе стабилизация сильно неопределенной системы; построение астатической системы слежения произвольного порядка; фильтрация и дифференцирование при неизвестной интенсивности шума; декомпозиция в многосвязных системах; оптимизация при неизвестных градиентах оптимизируемой функции и функций, задающих ограничения; основные задачи теории инвариантности; задачи управления при различного рода ограничениях и идентификации параметров динамических систем.

Теория СПС получила общесоюзное признание, в 1972 г. цикл работ по СПС удостоен Ленинской премии в области науки и техники.

Отметим, что на Западе теорию СПС "прозевали", ознакомление с ней произошло только в конце 1970-х — начале 1980-х гг. Ее популярность в мире растет и по сей день: огромное число публикаций, конференции и семинары посвящаются СПС и ее приложениям, в особенности, в робототехнике, электроприводе, авиации и космонавтике. Сегодня в учебных программах крупнейших университетов мира имеется раздел по теории СПС, и редкая конференция по управлению обходится без секции по СПС и скользящим режимам.

Академическая общественность оценила вклад С. В. Емельянова в науку: с 1967 г. он стал заместителем директора Института проблем управления (ИАТ) по науке, а в 1970 г. был избран членом-корреспондентом АН СССР.

На смену эпизодическим приложениям теории СПС в промышленности, оборонной технике, энергетике приходит понимание необходимости создания на базе СПС конкурентоспособного комплекса технических средств локальных информационно-управляющих систем (КТС ЛИУС). Такой комплекс СУПС, состоящий из более чем 40 приборов, был разработан под научным руководством С. В. Емельянова совместно с фирмой "Энергоинвест" (г. Сараево, Югославия) и серийно производился на заводе "Точэлектроприбор" (г. Киев, Украина). С середины 1970-х гг. КТС ЛИУС-СУПС был базовым инструментом для систем локальной автоматики в промышленности, энергетике и строительстве.

Параллельно с гражданским применением СПС велась работа в оборонной промышленности. За создание новых высокоэффективных образцов военной техники С. В. Емельянову в 1980 г. была присуждена Государственная премия в области науки и техники.

Научные интересы С. В. Емельянова весьма разнообразны и не ограничиваются теорией СПС, он активно разрабатывает проблемы комплексной автоматизации технических объектов, имитационного и глобального моделирования, вносит значительный вклад в системный анализ и теорию принятия решений. Особое его увлечение — экономика, главным образом — применение теории обратной связи для объяснения регуляторных механизмов рыночных экономик. Широко известен курс популярных лекций по саморегулирующимся экономическим механизмам, с которыми он выступал перед самыми разными аудиториями.

За реализацию крупного проекта комплексной автоматизации Запсибметкомбината, в котором нашли воплощение многие идеи С. В. Емельянова, группа специалистов получила в 1981 г. премию Совета Министров СССР в области науки и техники.

В 1976 г. С. В. Емельянов совместно с Д. М. Гвишиани организует Институт системных исследований (в дальнейшем Институт системного анализа — ИСА), ориентированный на выполнение крупных комплексных проектов, требующих междисциплинарного подхода.

В 1976 г. он переходит из ИПУ в ИСА и возглавляет работу по глобальному и имитационному моделированию, математическим методам информатики и системного анализа. В 1984 г. С. В. Емельянова избирают действительным членом АН СССР по отделению "Вычислительная техника, информатика и автоматизация".

Все годы С. В. Емельянов уделяет большое внимание международному сотрудничеству ученых, с конца 1960-х гг. он бессменный руководитель Международного коллектива ученых при ИПУ, а с 1976 г. — генеральный директор международного НИИ проблем управления (МНИИПУ). Этот институт сыграл огромную роль в координации усилий ученых разных стран по выполнению комплексных проектов социально-экономического, научно-технического и организационного характера. За большой личный вклад в эту работу академик С. В. Емельянов удостоен высших правительственных наград ряда стран.

В период работы над КТС ЛИУС-СУПС выявились ограничения по использованию скользящих режимов в технических системах и, в особенности, в системах прямого цифрового управления. Размышляя над возникшими проблемами, С. В. Емельянов пришел к выводу, что первопричина инвариантности движения в скользящем режиме — не сам скользящий режим (он служит лишь средством), а более фундаментальное явление — обратная связь нового типа, названная им впоследствии координатно-операторной. Развитие этой концепции в период с 1981 по 1991 г.

привело к формированию нового раздела современной теории обратной связи. Новая разработка существенно обогатила теорию управления новыми методами управления в условиях неопределенности, позволила вскрыть общий механизм синтеза законов управления при дефиците информации. Новую трактовку получили вопросы теории инвариантности, появилась возможность дать законченную теорию дискретных и цифровых СПС, были разработаны методы многократного помехозащищенного дифференцирования, фильтрации, разделения смеси сигналов с неизвестными характеристиками, оптимального управления при неопределенности, стабилизации сложных, в том числе, нелинейных систем. В частности, удалось вскрыть природу скользящих режимов и показать, что на самом деле это очень сложное, многоплановое явление, но что по определенному признаку (порядку скольжения) их можно упорядочить. Стандартный скользящий режим оказался одним из самых простых и был назван режимом скольжения первого порядка, а всего теоретически параметров скольжения бесконечно много. В 1994 г. цикл работ по теории новых типов обратной связи удостоен Государственной премии России в области науки и техники. Область научных интересов С. В. Емельянова охватывает проблемы нелинейной динамики и управления. Им написаны монографии по методам анализа глобальной управляемости и стабилизируемости нелинейных динамических систем, по геометрическим методам в вариационных задачах, фундаментальные труды по математическим методам нелинейного анализа. В последние годы С. В. Емельяновым получены важные результаты по робастной устойчивости, по управлению хаотической динамикой и по локализации неустойчивых циклов.

В поле зрения С. В. Емельянова постоянно находятся проблемы воспитания научной смены: он основал всемирно известную научную школу, у него много учеников и последователей, среди которых академики и члены-корреспонденты РАН, других академий, руководители институтов, фирм, кафедр университетов и академических лабораторий, более трех десятков лет он работал в ВАК. С. В. Емельянов плодотворно работает в системе высшего образования, основал кафедры по актуальным направлениям науки и техники: в Московском институте стали и сплавов — инженерной кибернетики (1964 г.), в Московском государственном университете им. М. В. Ломоносова — нелинейных динамических систем и процессов управления (1989 г.), в Московском физико-техническом институте — системного анализа (1985 г.). Эти кафедры выпустили сотни высококвалифицированных специалистов. Создание кафедры инженерной кибернетики в Московском институте стали и сплавов послужило стимулом к организации нового факультета информатики и экономики.

С. В. Емельянов — член редколлегий многих научных журналов, в том числе “Доклады РАН”, “Дифференциальные уравнения”, “Автоматика и телемеханика”, “Проблемы управления”, главный редактор журнала “Информационные технологии и вычислительные системы”.

С. В. Емельянов избирался членом бюро сначала Отделения механики и процессов управления РАН, а затем, после образования Отделения информатики, вычислительной техники и автоматизации РАН в 1983 г., членом бюро этого нового Отделения. С 1990 г. он являлся академиком-секретарем Отделения и членом Президиума РАН. Под его руководством Отделение превратилось в одно из крупнейших. По его инициативе и при непосредственном участии учеными институтов Отделения были сформированы государственные научно-технические программы в области информатики и информационных технологий, выполнены фундаментальные и прикладные исследования, имеющие огромное значение для развития промышленности, обороны, социальной сферы и других областей жизнедеятельности государства. В настоящее время он заместитель академика-секретаря Отделения информационных технологий и вычислительных систем и научный руководитель ИСА РАН и МНИИПУ. За большой вклад в науку и подготовку высококвалифицированных кадров он неоднократно удостаивался высших правительственных наград.

И в 75 лет Станислав Васильевич Емельянов полон сил и энергии, всецело настроен на активную продуктивную работу — только за последние пять лет им опубликовано четыре книги и десятки статей. Научная общественность высоко оценила достижения С. В. Емельянова за последнее пятилетие: ему присуждены премия РАН имени академика А. А. Андронова за вклад в теорию нелинейных динамических систем и Ломоносовская премия МГУ по науке I-ой степени за развитие математических методов качественного анализа сложных динамических систем. Неизменная доброжелательность, демократизм, готовность прийти на помощь, жизненная мудрость и принципиальность снискали ему любовь и уважение нескольких поколений ученых.

Желаем Станиславу Васильевичу многих лет жизни, насыщенной свершениями задуманного, здоровья и полного человеческого счастья.

Редакционный совет и
редакция журнала “Проблемы управления”

ЕВГЕНИЙ АЛЕКСАНДРОВИЧ ФЕДОСОВ
(к 75-летию со дня рождения)

14 мая 2004 года исполнилось 75 лет со дня рождения Героя Социалистического Труда, лауреата Ленинской и Государственной премий СССР, академика Российской академии наук, заместителя академика-секретаря Отделения энергетики, машиностроения, механики и процессов управления, председателя Секции проблем машиностроения и процессов управления, председателя Объединенного научного совета РАН по комплексной проблеме “Процессы управления и автоматизация”, крупнейшего российского ученого в области авиационного ракетостроения, члена редакционного совета журнала "Проблемы управления" Евгения Александровича Федосова.

Е.А. Федосов родился 14 мая 1929 года в г. Москве. Научная деятельность Евгения Александровича началась в 1952 г. после окончания Московского высшего технического училища им. Н.Э. Баумана, хотя его склонность к научным исследованиям ярко проявилась еще в студенческие годы. В 1956 г. он закончил аспирантуру МВТУ и успешно защитил диссертацию на соискание ученой степени кандидата технических наук.

В 1967 году Е.А. Федосов защитил диссертацию на соискание ученой степени доктора технических наук, а в 1969 г. ему присвоено ученое звание профессора.

С 1954 г. Е.А. Федосов работает в Государственном научно-исследовательском институте авиационных систем (ГосНИИАС), с которым была связана вся его дальнейшая творческая деятельность. В ГосНИИАС Е.А. Федосов прошел путь от инженера до директора Института, которым он стал в 1970 г.

С 2001 года Евгений Александрович – генеральный директор ФГУП “ГосНИИАС”. Под его руководством Институт стал головным исследовательским центром авиационной промышленности. Разработка и создание отечественной военной авиации страны, становление и развитие авиационно-ракетной техники неразрывно связаны с именем Е.А. Федосова. За большой личный вклад в создание самонаводящихся ракет и работы по управляемому авиационному оружию Е.А. Федосов награжден орденом “Знак почета” (1966 г.) и Орденом Ленина (1971 г.).

В эти же годы под его руководством и непосредственном участии создавались теоретические основы проектирования авиационных систем. Большой личный вклад Е. А. Федосов внес в разработку принципов построения самонаводящегося высокоточного авиационного оружия, которые были использованы при его проектировании и серийном производстве. За цикл работ по этим направлениям в 1976 г. Е. А. Федосову в составе группы ученых и конструкторов была присуждена Ленинская премия. В 1980-е годы под руководством и при его непосредственном участии разрабатывались стратегические самолеты-носители и крылатые ракеты стратегического назначения. Были проведены работы по исследованию, разработке и созданию авиационных корреляционно-экстремальных систем навигации с использованием цифровых карт местности, созданы методы обработки оптических и телевизионных изображений. По инициативе Евгения Александровича в ГосНИИАС были созданы уникальные моделирующие комплексы и осуществлена интеграция бортовых комплексов самолета-носителя и крылатых ракет. За эти работы Е. А. Федосову в 1983 г. было присвоено звание Героя Социалистического Труда.

С 1992 г. ГосНИИАС под руководством Е. А. Федосова активно проводил работы по внедрению наукоемких технологий в сферу гражданской авиации и другие конверсионные направления, модернизировал боевые самолеты и вертолеты, разрабатывал новые системы авионики, в том числе и для самолета нового поколения ИЛ-96 М/Т.

За разработку концепции боевого самолета пятого поколения и его систем вооружения и создание технологии виртуального прототипирования бортового радиоэлектронного оборудования в

2001 г. Евгению Александровичу присуждена премия Правительства Российской Федерации.

Е. А. Федосов активно участвует в разработке концепции и программы реструктуризации оборонно-промышленного комплекса России, является инициатором и одним из руководителей крупнейшей научно-технической Федеральной комплексной целевой программы "Национальная технологическая база", возглавляет экспертный совет программы.

Е. А. Федосов широко известен в научных кругах страны и за рубежом как крупный ученый в области теории систем со случайными параметрами, теории нестационарных систем управления летательных аппаратов, теории самонаведения и навигации. Он является основателем и лидером известной в мире научной школы, занимающей ведущую роль в области навигации и наведения летательных аппаратов, создания новых наукоемких технологий, обеспечивающих высокий уровень российской авиации.

Евгений Александрович является автором более 250 научных работ, изобретений и патентов, в том числе 11 монографий.

В 1979 году Е. А. Федосов избран членом-корреспондентом Академии наук СССР, а в 1984 году – академиком Академии наук СССР (ныне – Российская академия наук). За выдающиеся работы в области моделирования сложных технических систем в 1989 г. Е. А. Федосов награжден "Большой Золотой медалью им. акад. Б.Н. Петрова".

Евгений Александрович проводит большую научную и организационную работу, являясь заместителем академика-секретаря Отделения энергетики, машиностроения, механики и процессов управления, председателем Секции проблем машиностроения и процессов управления, возглавляет Объединенный научный совет по комплексной проблеме "Процессы управления и автоматизации", является главным редактором журнала "Известия РАН. Теория и системы управления", членом редакционных советов ряда журналов и научных изданий.

Редакционный совет журнала "Проблемы управления" сердечно поздравляет с 75-летним юбилеем Евгения Александровича Федосова, желает крепкого здоровья, многих лет жизни и счастья в его неутомимой деятельности.

IV Международная конференция и выставка

"Системы проектирования, технологической подготовки производства и управления этапами жизненного цикла промышленного продукта" (CAD/CAM/PDM-2004)

состоится 2 – 5 ноября 2004 г. в Институте проблем управления им. В.А. Трапезникова РАН

ТЕМАТИКА КОНФЕРЕНЦИИ:

- *организация структур технических и программных средств проектирования и управления; средства взаимодействия, структуры данных, международные стандарты;*
- *использование средств виртуальной реальности в промышленных системах;*
- *PDM-системы; интегрированные производственные системы и управление технологическими процессами;*
- *проектирование в машиностроении и строительстве;*
- *проектирование в радиоэлектронике.*

Телефоны для справок (095) 192-85-15, 192-85-43, 334-93-50, факс (095) 334-91-29.
E-mail: conf18@ipu.rssi.ru.

ТЕКУЩАЯ ИНФОРМАЦИЯ (формы заявок, правила оформления тезисов, докладов и т. п.):
http: //lab18.ipu.rssi.ru

CONTENTS & ABSTRACTS

FUZZY CONTROLLERS AND CONTROL SYSTEMS 2

Kudinov Yu. I., Dorokhov I. N., Pashchenko F. F.

Analysis, synthesis and classification problems of fuzzy controllers and control systems are discussed. The publications on fuzzy controllers are analyzed with the focus on logic-linguistic and taught controllers and control systems. A fuzzy difference TS-model, which describes dynamic objects and fuzzy controllers, is considered. The application of Lyapunov techniques for fuzzy systems stability estimation and fuzzy controller synthesis is shown. System structure and teaching techniques for fuzzy control systems with neuro-fuzzy TS plant and controller models are presented. Fuzzy and statistical controllers and control systems are compared.

NONSTATIONARY PROCESS MONITORING: ANALYSIS AND INVESTIGATION OF STEADY STATE CHANGES 15

Grebenyuk E. A.

The algorithm of on-line steady state change detection is offered, which detects the switches from nonstationary process to stationary and vice versa. The analysis of process properties influence on cointegration relationships is undertaken.

MULTI-OBJECTIVE INVESTMENT PORTFOLIO OPTIMIZATION. THE DETERMINISTIC CASE. . . .21

Rykov A. S., Iskhodjanov R. R.

The paper offers a method for solving a multi-objective constrained optimization problem of resource allocation that takes place in investment portfolio management. The method belongs to the deformed configuration methods class. Some problems of the method application are also considered.

INFORMATIONAL MANAGEMENT IN SOCIO-ECONOMIC SYSTEMS: CONTROL ELEMENTS AND THE WAYS OF INFORMATIONAL INFLUENCE 25

Kononov D. A., Kulba V. V., Shubin A. N.

The development results of a topical branch of research — informational management — are submitted. The paper provides the structural analysis of a socio-economic system and describes the mechanisms and ways of informational management.

CORPORATION MANIPULATING MECHANISMS. 34

Agheev I. A., Gurlev I. V., Tsyganov V. V.

Conditions for progressiveness and optimality of corporation evolution management mechanisms with the self-learning of classification in stochastic environment are found. The analysis of the manipulation mechanisms used in process of technocratic management of organization's evolution is undertaken. Two types of manipulation are considered the first type manipulation effects the emotions of the interested person (Pupil), while the second type employs quantitative indices and the rationality of the self-learning interested person (Scientist). The paper shows that there exists such manipulation, under which the corporative mechanism becomes regressive.

STRUCTURAL AND PARAMETRIC IDENTIFICATION OF ABNORMAL SITUATIONS IN COMPLEX PROCESS SYSTEMS 39

Ivashkin Yu. A.

The paper discusses the matrix models of structurally complex situations in large systems interaction and the algorithms for their identification and forecasting.

DESIGN CONCEPTS OF INTELLIGENT USER INTERFACE FOR OPERATOR SUPPORT SYSTEMS. 43

Lebedev V. G.

High-level architecture of an intelligent operator interface for controlling a complex hard-/software system is considered as well as the functionality of its key modules.

SYSTEM-LEVEL SELF-DIAGNOSIS OF ARISING FAULTS AND THEIR ELIMINATION IN RECONFIGURABLE SYSTEMS 48

Vedeshenkov V. A.

The paper offers a system-level method for self-diagnosis (SD) and elimination of arising faults of the components (modules and communication lines) of reconfigurable systems. The method assumes stable multiple faults of a limited number of components where a single fault may arise during the SD process. In order to detect such faults and to provide the correct diagnosis, the SD process is started two or more times, and the last two diagnoses are compared. The moving redundancy technique is used to eliminate the failed components by on-line reconfiguration. A case study of the SD method implementation for a single fault during the SD process and the reconfiguring of a 2-fault-tolerant system is also included.

DESIGN CONCEPTS OF ONBOARD CONTROL COMPLEXES FOR AUTOMATIC SPACECRAFTS 62

Mikrin E. A., Sukhanov N. A., Platonov V. N., et al.

The paper expounds an approach to onboard control complexes (OCC) design for automatic spacecrafts, and describes key OCC components and their tasks. It presents the structure of software, which is the key element in OCC integration. Finally, the OCC performance of Yamal telecommunication spacecrafts operated at geostationary orbit is considered.

MULTIPLICATIVE ALGEBRA AND ITS APPLICATION IN LOGICAL DATA PROCESSING 67

Vykhovanets V. S., Malyughin V. D.

The paper examines the problem of formulae synthesis for discrete functions based on multi-stage (decision making) and spectral decompositions carried out in multiplicative algebra. It shows the relationship between the synthesis of multiplicative forms and the generic problem of formulae synthesis, and makes practical recommendations on hardware and software implementation of the formulae obtained.

HARD REALTIME OPERATION OF LOCAL GRID SYSTEMS 78

Stetsyura G. G.

The application of grouped operations and data transfer/handling overlapping in local GRIDs is considered. This enables "hard realtime" tasks on such GRIDs.

S. V. YEMELYANOV (to the 75-th anniversary) 83

E. A. FEDOSOV (to the 75-th anniversary) 86