

681.5
Н 22

Министерство высшего и среднего специального
образования УССР

Харьковский Ордена Трудового Красного Знамени
институт радиозлектроники имени М. Я. Янгеля

АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ И ПРИБОРЫ АВТОМАТИКИ

В Ы П У С К 66

Республиканский
межведомственный
научно-технический
сборник

Основан в 1965 г.



ХАРЬКОВ
ИЗДАТЕЛЬСТВО ПРИ ХАРЬКОВСКОМ
ГОСУДАРСТВЕННОМ УНИВЕРСИТЕТЕ
ИЗДАТЕЛЬСКОГО ОБЪЕДИНЕНИЯ „ВИЩА ШКОЛА“
1963

Автоматизированные системы управления и приборы автоматики: Респ. междувед. науч.-техн. сб. Вып. 66. — Харьков: Вища школа, Изд-во при Харьк. ун-те, 1983. — 136 с.

В сборнике представлены результаты исследования проблемы построения автоматизированных систем управления с учетом задач автоматизации проектирования сложных систем. Рассмотрены теоретические вопросы кибернетики, адаптации и распознавания в системах управления и контроля, вопросы алгоритмизации в задачах линейного и динамического программирования. Обсуждены особенности применения методов оптимизации при решении различных практических задач.

Нормативные материалы приведены по состоянию на 1 января 1983 г. Для научных работников и специалистов.

Редакционная коллегия: Е. П. Путятин (отв. ред.), В. Ф. Шостаков (зам. отв. ред.), Н. А. Лысенков, (отв. секр.), В. И. Васильев, Э. А. Дедюков, В. В. Дубровский, А. Г. Евдокимов, И. В. Кузьмин, С. Ф. Капалап, Е. Н. Лазаренко, Н. Д. Коноваленко, В. И. Мелешко, В. М. Михайленко, И. Ф. Огороднейчук, Э. Г. Петров, Л. А. Растрини, В. В. Свиридов, Ю. Г. Стоян, И. Н. Щищев

*Адрес редакционной коллегии: 310059, Харьков-59, пр. Ленина, 14. Институт радиоэлектроники, тел. 40-94-59

Редакция естественнонаучной литературы

Ю. Н. МИНАЕВ, д-р техн. наук, И. С. ЧЕРЕПИНА,
М. Л. ГУРАЛЬНИК

К ВОПРОСУ ПОСТРОЕНИЯ АЛГОРИТМОВ РЕШЕНИЯ ЗАДАЧ ЦЕЛОЧИСЛЕННОГО ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ В УСЛОВИЯХ ВЛИЯНИЯ ВОЗМУЩЕНИЙ

Многие оптимизационные задачи АСУ математически формулируются как задачи целочисленного линейного программирования в условиях влияния возмущений. Различного рода возмущения приводят к тому, что параметры расчетной модели задаются нечетко, интервально. Влияние возмущений отражается на коэффициентах как матрицы ограничений, так и целевой функции.

В этом случае необходимо оценить инвариантность оптимального решения или определить максимально допустимый уровень возмущений, при котором решение сохраняется оптимальным.

Реальные прикладные задачи имеют большую размерность, и это затрудняет нахождение их оптимальных решений даже при отсутствии возмущений. Из формальных методов для решения задач целочисленного линейного программирования используют метод Гомори (правильных отсечений) и его различные модификации. Результаты многократного решения таких задач характеризуются недостаточно высокой сходимостью, а метод ветвей и границ сложно реализуется на ЭВМ [1, 2]. Отсюда понятен интерес к приближенным методам решения экстремальных задач целочисленного программирования. Приближенные методы позволяют находить решение задач целочисленной оптимизации в допустимые сроки с практически достаточной точностью.

В работе предлагается алгоритм решения задачи целочисленного линейного программирования в условиях возмущения, отражающихся на коэффициентах линейной формы.

При построении вычислительной схемы решения используют методы, в основе которых лежат эвристические процедуры [3].

Задача линейного программирования с целочисленными переменными записывается так:

$$\text{максимизировать } L(x) = \sum_{i=1}^N c_i x_i \quad (1)$$

при условиях $x_i \geq 0$, $x \in X$ (2); $X = \{x | Ax \leq b\}$ (3); $x_i = \{0, 1, 2, \dots\}$, $i \in N$ (4).

При воздействии возмущений на коэффициенты линейной формы (при их интервальном представлении) задача (1) — (4) трансформируется в задачу (2) — (5)

$$L_1(x) = \sum_{i=1}^N (c_i \pm w_i) x_i, \quad (5)$$

где W_i — возмущения; $W = \{w_i\}$, $i = 1, \overline{N}$.

Если коэффициенты линейной формы заданы как счетное множество, то задача (1) — (4) преобразуется в (2) — (4), (6):

$$L_2(x) = \sum_{i=1}^N c_i^{(n)} x_i, \quad i = 1, 2, \dots \quad (6)$$

где $c_i^{(n)}$ — множество значений c , возникающих при возмущениях $c_i = [c_i^{(n)}]$, $i = 1, 2, \dots$, $i \in N$.

Решение задачи целочисленной оптимизации в условиях возмущений сводится к проверке инвариантности решений задач (1) и (5) или задач (1) и (6) при ограничениях (2) — (4).

Метод эвристической самоорганизации (МЭС) предполагает выполнение следующих эвристических процедур [3]: 1) выбор элементарных алгоритмов образования входных сигналов; 2) выбор эвристических критериев самоотбора решений; 3) выбор закона генерации комбинаций.

Алгоритмы образования входных сигналов являются наиболее мощным средством эвристической самоорганизации, позволяющим на отдельных шагах алгоритма сочетать различные методы получения входных сигналов (градиентные, случайные и др.). На основании (1) формулируются две задачи (прямая и обратная) линейного программирования без требования целочисленности (4).

Пусть решениями прямой задачи

$$\sum_{i=1}^N c_i x_i \rightarrow \max \quad (7)$$

и обратной

$$\sum_{i=1}^N c_i x_i \rightarrow \min \quad (8)$$

при ограничениях (2) — (3) являются векторы $x^{*(1)}$ и $x^{*(2)}$ соответственно.

На основании процедур последовательного деления отрезка $[x^{*(1)}, x^{*(2)}]$ формируются массивы целочисленных векторов, удовлетворяющих условиям (2) — (3). Процедуры половинного деления отрезка многократно повторяются на основании соотношения

$$\forall i, \tilde{x}_i^{(k)} = 0,5 [\tilde{x}_i^{(1)} + \tilde{x}_i^{(k-1)}] + \tilde{x}_i^{(k-1)}, \quad k = 1, 2, \dots \quad (9)$$

Условием окончания процедуры (9) является выполнение условий

$$\forall i, \quad |\hat{x}_i^{(i)} - x_i^{*(1)}| \leq \Delta, \quad i = \overline{1, N}, \quad (10)$$

где Δ — интервал дискретности.

На основании совокупности векторов $\hat{x}$ формируется последовательность целочисленных векторов $X1 \in X$ и $X2 \in X$ (путем округления и отбрасывания дробной части), представляющих собой опорную последовательность $\hat{X} = X1 \cup X2$. Векторы $\hat{x}^{(i)} \in \hat{X}$, $i = 1, 2, \dots$ являются входными сигналами для второго этапа МЭС. Пороговый самоотбор осуществляется путем выбора некоторого подмножества векторов $\hat{x}^{(i)}$ из всего множества $\hat{x}^{(i)}$ в зависимости от нормы этих векторов $r^{(i)} = \langle \hat{x}^{(i)}, \hat{x}^{(i)} \rangle^{1/2}$ и нормы вектора решения прямой задачи (7) линейного программирования без условия целочисленности $r^* = \langle x^{*(1)}, x^{*(1)} \rangle^{1/2}$.

Все векторы $\hat{x}^{(i)} \in \hat{X}$, $i = 1, 2, \dots$, удовлетворяющие условию $\forall j, \hat{x}_j^{(i)} \in \hat{x}_j$, $j \in N$ (11), и для нормы которых справедливо $r^{(i)} > \lambda r^*$, $j = 1, 2, \dots, j \leq l$ ($0,7 < \lambda < 1$), принимаются в качестве расчетных для третьего этапа МЭС. Для каждого вектора $\hat{x}^{(i)}$ формируется массив $X^{(i)}$ таким образом, чтобы элементы главной диагонали представляли собой p -ю составляющую вектора $\hat{x}^{(i)}$, $p = 1, 2, \dots$, изменяющуюся на значение константы $k_p = 0, 1, 2, \dots, p \in N$, $j = 1, 2, \dots \text{diag } X^{(i)} [x_p^{(i)} + k_p]$, $p \in N$, $j = 1, 2, \dots$ (12), при выполнении условия $x_p^{(i)} + k_p \in X$, где k_p — интервал дискретности; p — номер строки:

$$X^{(i)} = \begin{bmatrix} x_1^{(i)} + k_{11} \in X & x_2^{(i)} & \dots & x_n^{(i)} \\ x_1^{(i)} & x_2^{(i)} + k_{22} \in X & \dots & x_n^{(i)} \\ x_1^{(i)} & x_2^{(i)} & \dots & x_n^{(i)} + k_{nn} \in X \end{bmatrix} \quad (13)$$

Генерация комбинаций представляет процедуру формирования массивов $\{x^{(i)}\}$ (13), $j = 1, 2, \dots$, и определения значений линейной формы $L(x_p^{(i)})$, $p \in N$, $j = 1, 2, \dots$ для каждой строки (вектора) $X^{(i)}$. Для всех $X^{(i)} \in X$, для которых $|L(x_p^{(i)})| \geq \mu L(x^{*(1)})$ при $0,5 < \mu < 1$, $j = 1, 2, \dots, p \in N$, (14), генерируются новые массивы по способу (13) и при выполнении условия $\forall p, x_p^{(i)} + k_p \in X$ определяется наибольшее значение среди всех линейных форм: $\max \{L(x_p^{(i)})\}$, $p \in N$, $j = 1, 2, \dots$. Вектор $x_m^{(i)}$, для которого $L(x_m^{(i)}) = \max_{p \in N} \{L(x_p^{(i)})\}$, принимается в качестве решения целочисленной задачи линейного программирования (1) — (4).

Условия инвариантности оптимальных решений при влиянии возмущений — задача (5), можно сформулировать в следующем виде.

Если линейная форма исходной задачи ограничена сверху и снизу и существуют решения сформированных непрерывных задач максимизации (7) и минимизации (8), то можно определить соответствующие двойственные конусы $D(M^{(1)})$ и $D(M^{(2)})$ при решениях $x^{*(1)}$ и $x^{*(2)}$, для которых справедливо включение вектора функционала $(c_i \pm W_i)$ в двойственный конус [4,5]:

$$(c_i \pm W_i) \in Q_x = D(M^{(1)}) \cap (-D(M^{(2)})) \quad (15)$$

Вычислительная схема алгоритма имеет следующий вид:

1. Формирование из нечеткой задачи целочисленного линейного программирования при интервальном задании коэффициентов линейной формы $c_i^{\min} \leq c_i \leq c_i^{\max}$ множества четких задач. Формирование предельной матрицы C , обладающей свойством $C = C1 \cup C2$,

$$[C1], \text{ diag } C1 = \{c_i^{\min}\},$$

$$[C2], \text{ diag } C2 = \{c_i^{\max}\},$$

где каждая строка матрицы C является линейной формой четкой ЗЛП. Этот прием позволяет привести решение нечетко сформулированной задачи к решению $2N$ четких задач линейного программирования.

2. Выделение одной четкой задачи целочисленного линейного программирования вида (1) — (4) путем произвольного выбора k -й строки из матрицы C , $k = \overline{1, 2N}$. K -я строка принимается в качестве линейной формы задачи линейного программирования с четкими коэффициентами

$$\sum c_i^{(k)} x_i \rightarrow \text{Extr}, k = \overline{1, 2N}, c_i^{(k)} \in [C].$$

3. Формирование прямой (7) и обратной (8) задач линейного программирования (ЗЛП) без требований целочисленности.

4. Решение непрерывных ЗЛП симплекс-методом. Векторы решений $x^{*(1)}$ и $x^{*(2)}$ соответственно.

5. Выполнение процедуры половинного деления (9), проверка условий (2) — (3) и фиксирование полученных значений в последовательности x .

6. Проверка условия (10). При выполнении условия переход к п. 5, иначе — к п. 7.

7. Формирование целочисленных векторов $X1$ и $X2$ с использованием процедур $X1 = \text{Entier}(\hat{x})$; $X2 = \text{Okrugi}(\hat{x})$.

8. Проверка условия $\{x1_i\} \in X$, $\{x2_i\} \in X$, $\forall (i, j)$. Векторы $X1$ и $X2$, не удовлетворяющие условию (3), отбрасываются.

9. Проверка условий $X1 \neq \emptyset$, $X2 \neq \emptyset$. Если $X1 = \emptyset$ или $X2 = \emptyset$, это свидетельствует об отсутствии в исходной задаче целочисленного решения или о невозможности определения его при помощи данного алгоритма. Переход к п. 20, иначе — к п. 10.

10. Вычисление $r^* = (x^*, x^*)^{1/2}$. Если (1) задача на максимизацию, то определяется норма вектора $x^{*(1)}$, иначе — норма вектора $x^{*(2)}$.

11. Выделение векторов, удовлетворяющих условию (11):

$$\hat{x}^{(j)} \in \hat{X}, j = 1, \dots, d.$$

12. Формирование из векторов п. 11 массивов вида (13) по закону $\forall (p \neq q), x_{pq}^{(i)} = \hat{x}_p^{(i)}, p \in N, q \in N, \forall (p = q), x_{pq}^{(i)} = \hat{x}_p^{(i)} + k_p$, удовлетворяющих условию (12), где p — номер строки; q — номер столбца массива (13).

13. Для каждой вектор-строки массива (13) вычисляются линейные формы $L = \{L_t x_{pq}^{(i)}\}$, $t = \overline{1, Nd}, d$ — количество массивов вида (13).

14. Выделение векторов $x^{(c)}$, удовлетворяющих условию (14), $c = 1, t2$.

15. Выполнение процедуры формирования векторов п. 12 при выполнении условия (12), иначе — переход к п. 16.

16. Определение в массиве значений линейной формы $\{L_c\} \subset \{L_t\}$ $c = 1, t2$ вектора $x_m^{(c)}$, на котором $L(x_m^{(c)}) = \max \{L_c\}$ $c = 1, 2, \dots, t2$.

17. Построение двойственных конусов $D(M^{(1)})$ и $D(M^{(2)})$ по ограничениям, которые в точках решения непрерывных ЗЛП $x^{*(1)}$ и $x^{*(2)}$ представляются строгими равенствами $Ax^{*(1)} = b$ и $Ax^{*(2)} = b$.

18. Построение конуса допустимых вариаций $Q_x = D(M^{(1)}) \cap \cap (-D(M^{(2)}))$.

19. Проверка принадлежности векторов конусу Q_x , $c' \in [C1]$, $c' \in Q_x$ и $c'' \in [C2]$, $c'' \in Q_x$. При выполнении условия (15) решение инвариантно в малом, переход к п. 20.

20. Печать результатов и комментариев.

Решение ЗЦЛП будет называться инвариантным в малом, если в исходной и возмущенной задачах последовательности векторов сохраняются неизменными, т. е. существует общее допустимое решение. Вычислительная схема алгоритма приведена на рисунке.

Пример. Имеется задача целочисленного программирования $\max (x_1 + x_2), x \in X$,

$$X = \{x | 3x_1 + x_2 \leq 18, 2x_1 + 8x_2 \leq 45, 5x_1 - 8x_2 \leq 15, x \geq 0\} \quad (16)$$

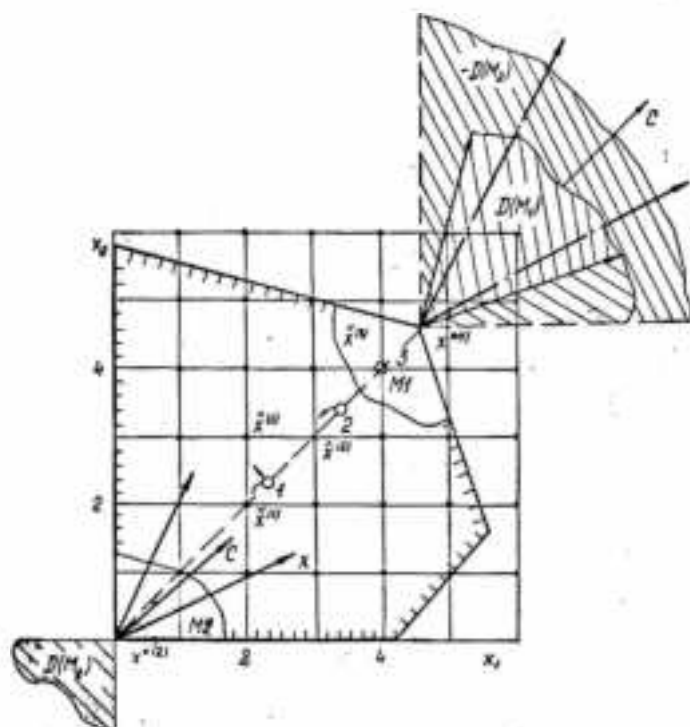
x_i — целые числа.

В условиях возмущения задача (16) трансформируется в следующее множество задач:

$$\begin{aligned} \max (x_1 + 2x_2), x \in X, \max (3x_1 + x_2), x \in X, \\ \max (x_1 + 4x_2) x \in X. \end{aligned} \quad (17)$$

Необходимо установить существование однозначного решения множества задач (17).

Алгоритм решения задачи 1°. Выделение исходной задачи ЛП при целочисленных переменных $\max(x_1 + x_2)$, $x \in X$.



Геометрическая интерпретация алгоритма решения ЦЗЛП при вариациях коэффициентов целевой функции

2°. Решения прямой и обратной задач без учета целочисленности представляются векторами $x^{*(1)} = [4,5; 4,5]$, $x^{*(2)} = [0; 0]$.

3°. Выполнение процедур половинного деления, формирование целочисленных векторов, определение норм векторов.

k	$\hat{x}(k)$	x_1	x_2	$r(x_1)$	$r(x_2)$
1	(2,22; 2,23)	(2,0; 2,0)	(2,0; 2,0)	2,82	2,82
2	(3,5; 3,55)	(3,0; 3,0)	(3,0; 4,0)	4,29	5,0
3	(3,95; 3,95)	(3,0; 3,0)	(4,0; 4,0)	4,24	5,69

4°. При $\lambda = 0,7$ и $r^{*(1)} = 6,35$ для реализации следующих шагов алгоритма отбираются векторы $X2^{(2)} = [3,0; 4,0]$ и $X2^{(3)} = [4,0; 4,0]$.

5°. Покоординатное изменение векторов $X^{(2)}$ и $X^{(3)}$ на единицу приводит к массивам:

$$X^{(1)} = \begin{vmatrix} 3+1 & 4 \\ 3 & 4+1 \end{vmatrix}$$

$$X^{(2)} = \begin{vmatrix} 4+1 & 4 \\ 4 & 4+1 \end{vmatrix}$$

6°. Проверка $x_p^j \in X$, $j = 1, 2$, выявляет, что $x_2^{(1)} = \{3; 5\}$, $x_1^{(2)} = \{5; 4\}$ и $x_2^{(3)} = \{4; 5\}$ не удовлетворяют условиям ограничений. Следовательно, вектор $x_1^{(1)} = \{4; 4\}$ должен приниматься в качестве решения задачи (16).

7°. Определение ограничений, образующих пьатры в точках $x^{*(1)}$ и $x^{*(2)}$: $M^{(1)}$ образован ребрами $3x_1 + x_2 = 18$; $2x_1 + 8x_2 = 45$. $M^{(2)} - x_1 = 0$; $x_2 = 0$.

8°. Построение конуса допустимых решений:

$$D(M^{(1)}) = \{(3, 1) (1, 4)\}; D(M^{(2)}) = \{(-1, 0) (0, -1)\}; Qx = \\ = \{(3, 1), (1, 4)\}.$$

9°. Проверка принадлежности направлений векторов (17) конусу Q , подтверждает, что множество возмущенных задач имеет общую опорную последовательность, т. е. решение является инвариантным в малом.

Список литературы: 1. Корбут А. А., Финкельштейн Ю. Ю. Дискретное программирование. — М.: Наука, 1969. — 368 с. 2. Емеличев В. А. Дискретная оптимизация, последовательные методы. — Кибернетика, 1971, № 6, с. 12—16, 1972, № 2, с. 31—37. 3. Ивахненко А. Г. Системы эвристической саморганизации в технической кибернетике. — К.: Техника, 1971. — 372 с. 4. Мишаев Ю. Н. Устойчивость оптимальных решений в задачах нелинейного математического выпуклого программирования. — В кн.: Автоматизированные системы управления предприятиями. Киев, 1977, с. 18—29. 5. Пшеничный Б. Н. Необходимые условия экстремума. — М.: Наука, 1969. — 112 с.

Поступила в редакцию 18.04.80.

УДК 681.142.4

Е. П. ПУТЯТИН д-р техн. наук, В. Б. КЛИМУШЕВ

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ПОНЯТИЯ ЗАДАЧИ

Сопоставление алгоритмов с оптимальными вычислительными комплексами предполагает выявление их характерных свойств.

Объединяет группу однотипных алгоритмов (т. е. алгоритмов, обладающих общими свойствами) понятие задачи, подле-

жащей решению. Оно имеет целевую направленность, т. е. решение задачи — цель, которую надо каким-либо образом достичь, причем пути достижения ее могут быть различны. Формализуем понятие задачи как категории, объединяющей группу однотипных алгоритмов. Для построения математической модели задачи воспользуемся понятиями формального языка и порождающей грамматики, терминология — работа [1].

Согласно [1] формальный язык (далее для краткости просто язык) представляет собой множество цепочек в некотором конечном алфавите, причем порождающей грамматикой (далее просто грамматикой) называется упорядоченная четверка, $G = (U, V_n, \sigma, P)$, где $U = \{a_1, a_2, \dots, a_m\}$ — основной (терминальный) алфавит; V_n — конечный вспомогательный (нетерминальный) алфавит (символы которого обозначаются малыми греческими буквами); $\sigma \in V_n$ — начальный (нетерминальный) символ; $P = \{u_i \rightarrow v_i \mid i = 1, 2, \dots, k\}$ — конечная система подстановок, левые и правые части которых есть цепочки $u_i, v_i \in F(v)$, где $F(v)$ — свободная полугруппа над объединенным алфавитом $V = (U \cup V_n)$, а символ $\rightarrow$ является внешним и не принадлежит алфавиту V .

Согласно [2,3] алгоритм характеризуется определенностью, результативностью, массовостью.

При анализе алгоритмов с точки зрения возможностей их реализации на ЭВМ особую роль играет свойство результативности. Для упрощения дальнейших рассуждений введем понятие вычислительного алгоритма, как категории абстрагирующей только свойство результативности.

Определение 1. Пусть имеем множество данных D и множество результатов R . Вычислительным алгоритмом a будем называть отображение $D \rightarrow R$.

Определение 2. Представлением $\Pi_a(L)$ вычислительного алгоритма a в языке $L(G)$, порожденным грамматикой G , будем называть некоторую правильную цепочку x , принадлежащую свободной полугруппе $F(U)$ над терминальным алфавитом U , причем $F(U) \supseteq R$, $F(U) \subset F(v)$ и $L(G) \supset R$.

Очевидно, что любому представлению Π_a можно поставить в соответствие множество $B_{L(G)}(\Pi_a)$ выводов $W_i(\Pi_a)$ цепочки x из цепочки y , где $y \in F(v)$. При этом

$B_{L(G)}(\Pi_a) = \{W_i(\Pi_a) \mid i = 1, 2, \dots, n\}$, $W_i(\Pi_a) = (y, z_1, z_2, \dots, z_n, x)$ и имеют место $y \Rightarrow z_1, z_{i-1} \Rightarrow z_i$ для $i (1 \leq i \leq n)$, $z_n \Rightarrow x$, а также $y \in D$. Вывод $W_i(\Pi_a)$ будем называть выводом представления.

Воспользуемся понятием длины вывода [4]. Тогда длиной $l(W_i)$ вывода $W_i = (z_1, z_2, \dots, z_n)$ будем называть количество непосредственных выводов, необходимых для реализации W_i .

Докажем следующее утверждение.

Утверждение 1. Для любого алгоритма a существует язык $L(G)$, порожденный грамматикой G , такой, что представление алгоритма a в языке L , а именно $\Pi_a(L)$, выводится за минимальное количество шагов, т. е. длина $W_i(\Pi_a)$ минимальна.

Доказательство. Рассмотрим множество языков, в которых алгоритм a является представимым. Легко показать, что мощность этого множества будет не менее чем счетной. Действительно, согласно [2], множество всех языков над конечным алфавитом имеет мощность континуума. В каждый язык рассматриваемого множества введем цепочку, представляющую алгоритм a и произведем соответствующие изменения в алфавите языка и порождающей грамматике. Тогда алгоритм a будет представимым в каждом языке из множества языков над конечным алфавитом, следовательно, мощность языков не менее чем счетная.

Обозначим множество языков, в которых представим алгоритм a , через M_a , причем $M_a = \{m_n | n = 1, 2, \dots\}$.

Рассмотрим множество $V_{m_n}(P_a)$ выводов представления алгоритма a , т. е. $P_a(m_n)$, в произвольном языке m_n , принадлежащем множеству M_a . Понятие длины вывода позволяет упорядочить множество $V_{m_n}(P_a) = \{W_i(P_a) | i = 1, 2, \dots, n\}$. Будем считать, что выполняется отношение $W_i \infty W_j$, если имеет место одно из следующих условий: $l(W_i) = l(W_j)$, $l(W_i) < l(W_j)$. Отношение ∞ является частичным порядком на множестве $V_{m_n}(P_a)$. Более того, для любой пары W_i, W_j либо $W_i \infty W_j$, либо $W_j \infty W_i$ и, следовательно, отношение ∞ образует линейный порядок на множестве $V_{m_n}(P_a)$, т. е. цепь. Значит, множество $V_{m_n}(P_a)$ имеет нижнюю грань, элемент которой обозначим $W_{i_n}(P_a)_{m_n}$. Будем называть множество $V_{m_n}(P_a)$ группой выводов в языке m_n .

Проанализируем теперь множество всех групп выводов $K = \{V_{m_n}(P_a) | n = 1, 2, \dots\}$.

Множество K можно частично упорядочить, используя отношение ∞ . В этом случае данное множество можно рассматривать следующим образом: $K = V_{m_1}(P_a) \cup V_{m_2}(P_a) \cup \dots \cup V_{m_n}(P_a)$.

Определим понятие нижнего конуса K^γ множества K как совокупность всех нижних граней множества $V_{m_n}(P_a)$, $n = 1, 2, \dots$. Очевидно, что будет существовать точная нижняя грань множества K , т. е. $\inf_n K$. Таким образом, существует элемент $W_{i_n}(P_a)_{m_n}$ такой, что $l(W_{i_n}(P_a)_{m_n}) \leq l(W_{i_n}(P_a)_{m_n})$ для любого n . Следовательно, существует язык m_n , в котором вывод представления алгоритма a , т. е. $P_a(m_n)$ происходит за минимальное количество шагов.

Следствие 1. Для группы алгоритмов $A = \{a_i | i = 1, 2, \dots, d\}$ существует грамматика G , порождающая язык $L(G)$, такой что представление в нем алгоритма a , т. е. $P_a(L)$, выводится за минимальное количество шагов.

Следствие 1 справедливо в силу того факта, что возможно объединить продукции грамматик G_i , удовлетворяющих утверждению 1, в одну систему, входящую в грамматику G .

Таким образом, задачей, объединяющей группу алгоритмов $A = \{a_i | i = 1, 2, \dots, d\}$, можно назвать грамматику G , порождающую язык $L(G)$, такую, что представление в нем любого алгоритма a_i из группы A , т. е. $P_{a_i}(L) \in L(G)$, выводится за минимальное количество шагов.

Рассмотренная математическая модель задачи может быть применена при исследовании алгоритмов нормализации изображений [5] в оптико-вычислительных системах в процессе сопоставления этих алгоритмов с оптимальными вычислительными комплексами.

Список литературы: 1. Глушко В. М., Цейтлин Г. Е., Ющенко Е. Л. Алгебра, языки, программирование. — К.: Наук. думка, 1974. — 173 с. 2. Марков А. А. Теория алгоритмов. — Тр. мат. ин-та им. В. А. Стеклова, АН СССР, 42, М.: Л., 1954, с. 50. 3. Крицкий Н. А., Миронов Г. А., Фролов Г. Д. Программирование и алгоритмические языки. — М.: Наука, 1975. — 214 с. 4. Гладкий А. В. Формальные грамматики и языки. — М.: Наука, 1973. — 240 с. 5. Пулятин Е. П. Теоретические предпосылки нормализации изображений. — Проблемы бионки, 1973, вып. 10, с. 82—86.

Поступила в редакцию 29.12.81.

УДК 519.713

В. А. ГОРОХОВАТСКИЙ

ВОПРОСЫ КЛАССИФИКАЦИИ ИЗОБРАЖЕНИЯ С ПРИМЕНЕНИЕМ АЛГОРИТМОВ НОРМАЛИЗАЦИИ

Математическую модель изображения на плоскости можно представить в виде функции [1]

$$B(x, y) = \varphi(g, a, \varepsilon, x, y), \quad (1)$$

где $B(x, y)$ — функция яркости изображения, заданная в некоторой области D плоскости (x, y) ; $B \in M$; M — множество классифицируемых изображений; g — параметр (возможно многомерный), определяющий детерминированное искажение изображения относительно эталона $B_0 \in M_0$; M_0 — множество эталонов; $M_0 \subset M$; a — параметр, задающий специфику изображения (признаки) или класс, к которому принадлежит функция B , а $a \in A$; ε — помеха, имеющая вероятностную природу.

Помеха определяет некоторый разброс реализаций изображения вокруг среднего значения, поэтому функция $B(x, y)$ представляет собой реализацию случайной функции с заданной плотностью распределения. Естественно предположить, что изменения изображения, вызванные существенными изменениями значений параметров g и a , не исчезают на фоне действия помех, иначе задача классификации усложняется и не допускает решения с малой вероятностью ошибки даже посредством человеческого зрения. Это соответствует ситуации, когда помехи имеют не слишком большую дисперсию, а в случае фоновых помех — незначительно влияют на определение параметров преобразования изображения, т. е. соотношение сигнал/шум достаточно велико. Тогда влияние помех

можно учитывать в статистических процедурах принятия решений о классе a , а модель (1) принимает вид $B(x, y) = \varphi(g, a, x, y)$ (2).

Рассмотрим случай, когда множество искажений представляет собой группу преобразований G с элементами $g \in G$. Тогда два изображения B_1 и B_2 рассматриваются как эквивалентные относительно G при фиксировании a , если существует преобразование $g \in G$, для которого $B_2 = B_1(g, a)$ или существует такое a , что $B_1, B_2 \subset m_a$, где $m_a = \{B_0^a, g \in G\}$; B_0^a — эталон класса a .

Группа является достаточно общей моделью множества преобразований. Так, в работе [2] показано, что если R — любое множество взаимно-однозначных отображений типа (2) пространства на себя, то классе всех конечных произведений отображений $r_1^{\pm 1}, r_2^{\pm 1}, \dots, r_m^{\pm 1}$ с $r_1, r_2, \dots, r_m \in R, m = 1, 2, \dots$, где каждый показатель степени равен $+1$ или -1 и среди элементов $r_1, r_2, \dots$ могут быть равные, образует группу преобразований. Таким образом, невырожденное преобразование координат плоскости $u = u(x, y), v = v(x, y)$ образует группу, если оно взаимно-однозначно.

Поясним действие группы преобразований на множестве изображений. Пусть $B(x, y)$ — изображение, носитель C_B которого (носить — замыкание множества $\{(x, y), B(x, y) \neq 0\}$) целиком содержится в поле зрения ω . Если G — группа преобразований плоскости $(x, y) \rightarrow g(x, y), g \in G$, то соответствующее преобразование изображения $B(x, y)$ определяется равенством $gB(x, y) = B(g^{-1}(x, y))$, где g^{-1} — преобразование, обратное g , и g принимает значения из множества преобразований $G_1 \in G$, для которых носитель C_{gB} принадлежит полю зрения ω : $C_{gB} = gC_B = \{(x, y) : g^{-1}(x, y) \in C_B\} \subset \omega$ (3).

Если отображение φ в (2) взаимно-однозначно отображает декартово произведение множества $G \times A \rightarrow M$, то на множестве изображений M каждой паре (g, a) соответствует единственное изображение $B \in M$. Однако практически встречаются ситуации, когда это условие не выполняется. Например, изображения, полученные поворотом плоского квадрата на углы $\pi/2$ и π , совпадают. Поэтому для группы преобразований G можно установить взаимно-однозначное отображение $\bar{\varphi}$ только между изображениями из класса эквивалентности m_a и элементами фактор-группы G/H группы G , где H — нормальный делитель группы $G, H = \{h\}, hB^1 = B^2, B^1 \subset m_a, h \in G$ [3]. Такое отображение, установленное для всевозможных $a \in A$, задает разбиение M на классы эквивалентности или орбиты элементов B_0^a относительно G

$$M = \bigcup_{a \in A} m_a. \quad (4)$$

Обозначим представитель фактор-группы G/H через g^* . Множество всех изображений m_a равносильно группе G/H .

Задача опознания инвариантного относительно группы мешающих преобразований G заключается в отнесении опознаваемого изображения к одному из классов эквивалентности m_a . Решающее правило при этом можно рассматривать как отображение $M \rightarrow A$. Будем считать функцию $\mathfrak{z}(B)$, осуществляющую это отображение, инвариантной по отношению к группе G , если при каждом a она принимает одно и то же значение независимо от G , т. е. если для любого $g \in G$

$$\mathfrak{z}(g(B)) = \mathfrak{z}(B) = a. \quad (5)$$

Наиболее благоприятная ситуация с точки зрения надежности распознавания — это случай, когда инвариантная функция на различных орбитах принимает различные значения. Построим такую функцию на основе оператора нормализации, применяемого к распознаваемому изображению.

Рассмотрим действие на множестве M оператора нормализации F [3], отображающего множество M в M_0 . В результате применения оператора F к изображению B имеем

$$F(B) = B_a^0, \text{ если } B \in m_a. \quad (6)$$

Применение оператора нормализации заключается в определении параметров преобразования g^* , т. е. в нахождении явного вида отображения Φ , и компенсации вычисленного преобразования.

Обычно отображение Φ задается множеством функционалов $\Phi_1(B), \Phi_2(B), \dots, \Phi_n(B)$, которые характеризуют преобразование g . Аналогичное множество функционалов находят и для эталонных изображений. Параметры преобразования определяются из уравнения нормализации, связывающего системы функционалов распознаваемого и эталонного изображений.

В силу свойства (6) результат применения оператора F не зависит от преобразований изображения из группы G . Поэтому можно использовать решающую функцию, основанную на сравнении нормализованного изображения с эталонами. Можно добиться, чтобы такая функция, кроме инвариантности, принимала различные значения на различных орбитах элементов из M_0 . Для этого значения метрики нормируются некоторым образом. Изображение B будет отнесено к классу с параметром a_1 , если функционал $I = \rho(F(B), B_{a_1}^0)$ (7) достигает экстремума при $a = a_1$, где ρ — метрика, введенная на множестве M .

Для обеспечения надежности инвариантной процедуры (7) при воздействии помех обычно применяют интегральные по полю зрения метрики ρ , где учитываются различия изображений по точкам или отдельным фрагментам. При этом возможна логическая обработка сходства фрагментов для более достоверного определения значения ρ . Многие методы сравнения с эталонами, реализующие выражение (7), допускают интерпретацию в рамках теории статистических решений [4]. Так, минимум вероят-

ности ошибки при распознавании (правило Байеса) достигается при минимизации расстояния

$$I = \frac{\min}{B_0 \in M_0} \sum_{i,j=1}^n |B(i, j) - B_0(i, j)|$$

между бинарными дискретными изображениями $B(i, j)$ и $B_0(i, j)$; $B_0(i, j) \in M_0$; $i, j = \overline{1, n}$. Решение по максимуму функции правдоподобия для аналогичных изображений принимается согласно метрике $\rho(B, B_0) = \sum_{i,j=1}^n (B(i, j) - B_0(i, j))^2$. При этом в качестве $B(i, j)$ выступает нормализованное изображение.

Рассмотрим подробнее процедуру применения оператора нормализации в случае, когда изображение, подлежащее классификации, может принадлежать одному из классов m_a , $a \in A$. Предположим для наглядности, что $A = \{a_1, a_2\}$. Так как класс эталонов M_0 формируется, вообще говоря, из произвольных представителей m_{a_1} , m_{a_2} , то при построении отображения $\overline{\Phi}$, как правило, посредством множества функционалов $\Phi_1, \Phi_2, \dots, \Phi_n$ [3] возможны следующие варианты выбора эталонов:

1. $\Phi_i(B_0^{a_1}) = \Phi_i(B_0^{a_2})$. 2. $\Phi_i(B_0^{a_1}) \neq \Phi_i(B_0^{a_2})$, $i = \overline{1, n}$.

Более общим является случай 2, который соответствует произвольному выбору эталонов. Для реализации этого варианта сначала применяется оператор нормализации в предположении, что распознаваемое изображение принадлежит классу m_{a_1} , затем вычисляется сходство нормализованного изображения с эталоном $B_0^{a_1}$. После этого к исходному изображению применяется оператор нормализации с эталонными параметрами из класса m_{a_2} и вычисляется сходство преобразованного изображения с эталоном $B_0^{a_2}$. По максимуму сходства принимается решение о классе изображения. Таким образом, случай 2 связан с возможным многократным по количеству классов применением алгоритмов нормализации, что при большом многообразии возможных эталонов приводит к громоздким вычислениям.

Преимуществом с этой точки зрения обладает случай 1. В силу взаимно-однозначного соответствия $M \rightarrow G/H$ выбор эталонов всегда можно произвести так, чтобы выполнялось условие 1, несмотря на то, что функционалы, применяемые для нормализации, часто имеют свои особенности и применимы не для произвольных изображений из M . Переход в пространстве эталонов при этом достигается путем однократного применения оператора нормализации.

Положительная черта ситуации 2 — это возможность произвольного выбора эталонов исходя, например, из соображений лучшей помехозащищенности и надежности алгоритма.

Очевидно, что наиболее общая ситуация, когда для некоторых подмножеств из M_0 выполняется условие 1, сводится к случаю 2. Оператор нормализации в таком случае необходимо применять столько раз, сколько различных наборов функционалов $\Phi_1, \Phi_2, \dots, \Phi_n$ существует для множества M_0 . Каждый раз вычисляется сходство нормализованного изображения со всеми эталонами, для которых значения функционалов совпадают.

Результаты экспериментального исследования быстродействия и помехозащищенности алгоритмов нормализации описаны нами в работе [5].

Список литературы: 1. Шибанов Г. П. Распознавание в системах автоконтроля. — М.: Машиностроение, 1973. — 424 с. 2. Кузнецов В. П. Инвариантность решений по отношению к мешающим параметрам. — Проблемы передачи информации, 1971, 7, вып. 4, с. 38—44. 3. Путькин Е. П. Теоретические предпосылки нормализации изображений. — Проблемы бионики, 1972, вып. 10, с. 82—89. 4. Дуда Р., Харт П. Распознавание образов и анализ сцен. — М.: Мир, 1976. — 511 с. 5. Путькин Е. П., Гороховатский В. А. Исследование алгоритмов нормализации при инвариантной классификации изображений. — АСУ и приборы автоматики, 1982, вып. 64, с. 12—15.

Поступила в редколлегию 19.02.82.

УДК 510.62

Н. Г. САРНАВСКИЙ, канд. техн. наук, Ю. П. ШАБАНОВ-КУШНАРЕНКО, д-р техн. наук

ОБ ИДЕНТИФИКАЦИИ МЕТОДОМ СРАВНЕНИЯ ДИНАМИЧЕСКИХ СИСТЕМ, ОПИСЫВАЕМЫХ ИНТЕГРАЛОМ СВЕРТКИ

В статье сформулирована и доказана теорема об экспериментально проверяемых методом сравнения необходимых и достаточных условиях, при которых динамическая система может быть математически описана интегралом свертки.

Пусть $X(-\infty, t]$, где t — какое-нибудь действительное число; X — линейное пространство почти всюду ограниченных на $(-\infty, t]$ измеримых функций. Более точно, $X(-\infty, t]$ — линейное пространство классов эквивалентности по равенству почти всюду функций x , для каждой из которых существует число $c > 0$ (вообще говоря, зависящее от функции) такое, что $|x(\tau)| \leq c$ почти при всех τ . Обозначим через K_t положительный конус в пространстве X . Рассмотрим систему предикатов $\Phi_t(x, y)$ (t — параметр), каждый из которых при соответствующем t определен на $K_t \times K_t$ и удовлетворяет условиям 1) рефлексивности; 2) симметричности; 3) транзитивности. Нас интересуют условия, при которых для всех $t \in (-\infty, \infty)$ и всех x и y имеет место равенство

$$\Phi_t(x, y) = D \left(\int_{-\infty}^t A(t-\tau) x(\tau) d\tau, \int_{-\infty}^t A(t-\tau) y(\tau) d\tau \right), \quad (1)$$

где D — предикат равенства; $A(\xi)$ — почти всюду неотрицательная функция из пространства $L[0, \infty)$.

Для формулировки этих условий нам понадобятся некоторые обозначения. Введем их. Для любой функции $x \in X[-\infty, t]$ и положительного числа ξ положим $\bar{x}_\xi(\tau) = x(\tau - \xi)$ (2). Таким образом, функция $\bar{x}_\xi$ является сдвигом функции x вправо на величину ξ . Далее, положим при любом $T > 0$

$$\bar{x}_T(\tau) = \begin{cases} 0, & t - T < \tau < t, \\ x(\tau), & \tau \leq t - T. \end{cases} \quad (3)$$

Теорема. Для того чтобы для семейства предикатов $\Phi_t(x, y)$ нашлась функция A , принадлежащая положительному конусу пространства $L[0, \infty)$, нормированная на единицу, и такая, что имеет место равенство (1), необходимо и достаточно, чтобы это семейство удовлетворяло следующим условиям: 4) для любых $t \in (-\infty, \infty)$ и $x, x', y, y' \in K_t$ из равенств $\Phi_t(x, x') = 1$ и $\Phi_t(y, y') = 1$ следует, что $\Phi_t(x + y, x' + y') = 1$; 5) Для любого $t \in (-\infty, \infty)$ и любого $x \in K_t$ существует единственное неотрицательное число $[fx](t)$ такое, что $\Phi_t(x[f x](t)) = 1$ (4) (б) (4) тем же символом $[fx](t)$ обозначена функция на $(-\infty, t]$, тождественно равная числу $[fx](t)$; 6) величина $[fx](t)$ как функция от x непрерывна в метрике $L(-\infty, t]$; 7) Для любой функции $x \in K_t$ $\lim_{T \rightarrow \infty} [\bar{x}_T](t) = 0$; 8) для любого $t \in (-\infty, \infty)$, любых $x, y \in K_t$ и любого положительного ξ из равенства $\Phi_t(x, y) = 1$ вытекает равенство $\Phi_{t+\xi}(x_\xi, y_\xi) = 1$.

Доказательство. Достаточность. Зафиксируем t и рассмотрим функции из K_t , для которых

$$\int_{-\infty}^t x(\tau) d\tau < \infty, \quad (5)$$

т. е. мы рассматриваем функции x , принадлежащие положительному конусу пространства $L(-\infty, t]$. Покажем, в первую очередь, что доказываемое утверждение справедливо для этого класса функций. Пусть x, y — произвольные функции из этого класса. Тогда из (4) имеем $\Phi_t(x, [fx](t)) = 1$, $\Phi_t(y, [fy](t)) = 1$. Поэтому условие 4) дает $\Phi_t(x + y, [fx](t) + [fy](t)) = 1$ (6). Но из условия 5) вытекает, что вещественным неотрицательным числом c таким, что $\Phi_t(x + y, c) = 1$, является число $[f(x + y)](t)$. Поэтому из (6) следует, что $[f(x + y)](t) = [fx](t) + [fy](t)$ (7).

Это означает, что $[fx](t)$ — аддитивный функционал на положительном конусе пространства $L(-\infty, t]$. Из условия 6) вытекает, что этот функционал непрерывный. Но в пространстве $L(-\infty, t]$ положительный конус является воспроизводящим, т. е. для всякого $x \in L(-\infty, t]$ найдутся неотрицательные функции y и z из этого пространства такие, что $x = y - z$. Поэтому

функционал $[f x](t)$ однозначно продолжается до аддитивного, непрерывного, а следовательно, линейного функционала на $L(-\infty, t]$. Тогда найдется по теореме об общем виде линейного функционала на $L(-\infty, t]$ почти всюду ограниченная на полуоси $(-\infty, t]$ функция $A_t(\tau)$ такая, что

$$[f x](t) = \int_{-\infty}^t A_t(\tau) x(\tau) d\tau \quad (8)$$

для любого x из положительного конуса пространства $L(-\infty, t]$. Поскольку равенство (8) справедливо для любого t и любого $x(\tau)$ из положительного конуса пространства $L(-\infty, t]$, то, заменяя t на $t + \xi$, для любой $\bar{x}(\tau)$ из положительного конуса пространства $L(-\infty, t + \xi]$ получаем $[f \bar{x}](t + \xi) = \int_{-\infty}^{t+\xi} A_{t+\xi}(\tau) \bar{x}(\tau) d\tau$.

В частности, для любой функции и любого положительного числа ξ для функции $\bar{x}_\xi$, связанной с функцией x равенством (2), имеем

$$[f \bar{x}_\xi](t + \xi) = \int_{-\infty}^{t+\xi} A_{t+\xi}(\tau) x(\tau - \xi) d\tau. \quad (9)$$

Далее, согласно (4) $\Phi_{t+\xi}(\bar{x}_\xi, [f \bar{x}_\xi](t + \xi)) = 1$. Кроме того, из [4] и условия 8) следует, что $\Phi_{t+\xi}(\bar{x}_\xi, ([f x](t))_\xi) = 1$. Сравнивая два последних равенства, с помощью условий 2) и 3) находим $[f \bar{x}_\xi](t + \xi) = ([f x](t))_\xi$ (10). Поскольку $[f x](t)$ является постоянной функцией от τ , то ее сдвигка согласно формуле (2) является той же константой для полуоси $(-\infty, t + \xi)$. Поэтому из (8) следует, что

$$([f x](t))_\xi = \int_{-\infty}^t A_t(\tau) x(\tau) d\tau. \quad (11)$$

Подставляя формулы (9), (11) в (10), получаем $\int_{-\infty}^{t+\xi} A_{t+\xi}(\tau) \times$
 $\times x(\tau - \xi) d\tau = \int_{-\infty}^t A_t(\tau) x(\tau) d\tau$.

Сделаем в левом интеграле замену переменной интегрирования по формуле $\eta = t + \xi - \tau$, а в правом — по формуле $\eta = t - \tau$ получаем

$$\int_0^\xi A_{t+\xi}(t + \xi - \eta) x(t - \eta) d\eta = \int_0^\xi A_t(t - \eta) x(t - \eta) d\eta. \quad (12)$$

Положим $y(\eta) = x(t - \eta)$. Поскольку $x(\tau)$ — произвольная неотрицательная функция из $L(-\infty, t]$, то $y(\eta)$ — произвольная неотрицательная функция из $L[0, \infty)$. Равенство может быть переписано в виде

$$\int_0^{\infty} \alpha(\eta) y(\eta) d\eta = 0, \quad (13)$$

где $\alpha(\eta) = A_{t+\xi}(t+\xi-\eta) - A_t(t-\eta)$.

Равенство (13) означает, что функция α ортогональна к положительному конусу пространства $L[0, \infty)$. Поскольку этот конус является воспроизводящим, то отсюда следует, что функция α ортогональна всему пространству. Следовательно, она является нулевым элементом сопряженного пространства. Итак, для всех t , всех положительных ξ почти при всех $\eta \in [0, \infty)$ будет $A_{t+\xi}[t+\xi-\eta] = A_t[t-\eta]$. В частности, при $t=0$ $A_\xi(\xi-\eta) = A_0(-\eta)$ (14). Заметим, что функция α является элементом пространства почти всюду ограниченных на $[0, \infty)$ функций, являющегося сопряженным к $L[0, \infty)$. Под ортогональностью элемента сопряженного пространства g к элементу исходного пространства x понимается равенство нулю функционала g на элементе x .

Положим $A(\eta) = A_0(-\eta)$ ($\eta > 0$). Тогда, заменяя переменную η переменной $\tau = \xi - \eta$, из (14) получаем при всех ξ $A_\xi(\tau) = A(\xi - \tau)$. Это равенство выполняется почти при всех τ . Поскольку интеграл (8) не зависит от значений функции A_t на множестве меры нуль, мы можем переписать равенство (8) в виде

$$[fx](t) = \int_{-\infty}^t A(t-\tau) x(\tau) d\tau. \quad (15)$$

Проверим, что функция $A(\xi)$ ($\xi \geq 0$) является неотрицательной всюду за исключением, быть может, множества меры нуль. Пусть S — множество точек, в которых A принимает отрицательные значения, и пусть мера этого множества не равна нулю. Зафиксируем какие-либо t ; $S_t = \{\tau / \tau = t - u, u \in S\}$. Положим

$$x(\tau) = \begin{cases} 1, & \tau \in S_t, \\ 0, & \tau \in (-\infty, t] \setminus S_t. \end{cases}$$

Очевидно, $x \in K_t$. Тогда из [15] следует, что $[fx](t) = \int A(u) du$. Следовательно, $[fx](t) < 0$, что противоречит условию 5).

Покажем теперь, что L — норма функции A равна 1. Поскольку эта функция неотрицательна, то нужно показать, что

$$\int_0^1 A(\xi) d\xi = 1. \quad (16)$$

Пусть $X_0(t)$ — функция, тождественно равная 1. В силу условия 1) $\Phi_t(x_0, x_0) = 1$ (17). Но по условию 5) единственной постоянной функцией, для которой справедливо (7), является функция $[fx_0](t)$. Следовательно, $[fx_0](t) = x_0$. В силу (15) тогда

$$\int_0^1 A(\xi) d\xi = \int_{-\infty}^1 A(t-\tau) d\tau = 1.$$

Отсюда, в частности, следует, что $A \in L[0, \infty)$.

Перейдем теперь к общему случаю: $x(\tau)$ — произвольная, т. е., не обязательно суммируемая на $(-\infty, t]$, функция из K_t . Заметим, что равенство (7) доказано без предположения о суммируемости функций x и y . Представим $x(\tau)$ в виде суммы $x(\tau) = x_T(\tau) + \bar{x}_T(\tau)$, где T — произвольное действительное число. Из (7) тогда следует $[f x](t) = [f x_T](t) + [f \bar{x}_T](t)$ (18). Но функция $x_T(\tau)$ является ограниченной, измеримой и ее носитель компактен. Поэтому она принадлежит $L(-\infty, t]$ и для нее, следовательно, справедлива

формула (8), т. е. $[f x_T](t) = \int_{-\infty}^t A(t-\tau) x_T(\tau) d\tau = \int_{t-T}^t A(t-\tau) \times$
 $\times x(\tau) d\tau$. Поэтому существует $\lim_{T \rightarrow \infty} [f x_T](t) = \int_{-\infty}^t A(t-\tau) x(\tau) \times$
 $\times d\tau$. Перейдем в (18) к пределу при $T \rightarrow \infty$. Воспользовавшись последним равенством и условием 7), получим

$$[f x](t) = \int_{-\infty}^t A(t-\tau) x(\tau) d\tau. \quad (19)$$

Пусть $x, y \in K_t$ и для них справедливо равенство $\Phi_t(x, y) = 1$ (20). Из условий 2), 3), 5) тогда следует, что $\Phi_t(y, [f x](t)) = 1$ (21). Но в силу условия 5) единственной постоянной функцией, удовлетворяющей такому условию, является $[f y](t)$. Следовательно, $[f y](t) = [f x](t)$ (22).

Пусть обратно выполняется (22). Это равенство вместе с (4) дает $\Phi_t(x, [f y](t)) = 1$. Но $\Phi_t(y, [f y](t)) = 1$. Из двух последних равенств и условий 2) и 3) вытекает (20). Таким образом, для всех $x, y \in K_t$ равенства (20), (22) выполняются или не выполняются одновременно. Это и означает справедливость равенства (1).

Необходимость. Пусть для семейства предикатов $\Phi_t(x, y)$ при всех $t \in (-\infty, \infty)$ и любых $x, y \in K_t$ справедлива формула (1). Проверим выполнимость условий 4) — 8). Справедливость 4)

следует из аддитивности функционала $\int_{-\infty}^t A(t-\tau) x(\tau) d\tau$. Для справедливости 5) достаточно показать, что для любой функции $x \in K_t$ существует единственное положительное число c , удовлетворяющее уравнению $\int_{-\infty}^t A(t-\tau) x(\tau) d\tau = \int_{-\infty}^t A(t-\tau) c d\tau$. Очевидно, это так. Поскольку функция A нормирована на единицу, то решение этого уравнения имеет вид

$$[f x](t) = \int_{-\infty}^t A(t-\tau) x(\tau) d\tau. \quad (23)$$

Поскольку $[f x](t)$ — линейный функционал на $L(-\infty, t]$, то условие 6) выполняется. Далее, для любой функции $x \in K_t$ $[f x_T](t) =$
 $= \int_{t-T}^t A(t-\tau) x(\tau) d\tau$. Поэтому справедливо 7). Наконец, ра-

венство $\Phi_t(x, y) = 1$ означает, что

$$\int_{-\infty}^t A(t-\tau) x(\tau) d\tau = \int_{-\infty}^t A(t-\tau) y(\tau) d\tau, \quad (24)$$

а равенство $\Phi_{t+\xi}(\bar{x}_\xi, \bar{y}_\xi) = 1$ означает

$$\int_{-\infty}^{t+\xi} A(t+\xi-\tau) x(\tau-\xi) d\tau = \int_{-\infty}^{t+\xi} A(t+\xi-\tau) y(\tau-\xi) d\tau. \quad (25)$$

Для того чтобы удостовериться, что из справедливости (24) вытекает справедливость (25), достаточно в (24) ввести новую переменную интегрирования $\eta = \tau + \xi$. Теорема доказана.

Поступила в редколлегию 20.01.81.

УДК 519.81

В. Е. ЖУКОВИН, канд. техн. наук, Ф. В. БУРШТЕИН,
К. Г. МУСХЕЛИШВИЛИ, канд. техн. наук, Г. И. СТОП-
ЧЕНКО

ОБ ОДНОМ КЛАССЕ МНОГОКРИТЕРИАЛЬНЫХ ФУНКЦИЙ ПОЛЕЗНОСТИ

При проектировании и функционировании сложных целенаправленных систем, к которым можно отнести автоматизированные системы управления, возникает задача неоднозначного выбора лучших решений (проектов, вариантов, альтернатив). При этом необходимо учитывать цели и желания лиц, принимающих решения (ЛПР), что приводит к необходимости исследования существования функции полезности и вычисления ее значений для заданных предпочтений ЛПР.

Задача формулируется следующим образом. Имеется множество возможных решений X , на котором некоторым образом задано отношение предпочтения $\geq$. Необходимо построить на X числовую функцию f , согласованную с этим отношением предпочтения, т. е. $f: X \rightarrow R^1$ и для всех $x, y \in X$ $x \geq y \Leftrightarrow f(x) \geq f(y)$ (1). Функция f и представляет собой искомую функцию полезности. Очевидно, что в результате построения такой функции выбор оптимального решения сведется к нахождению тех аргументов функции $f(\cdot)$, на которых она принимает свои экстремальные значения.

В известной теореме Кантора о необходимых и достаточных условиях существования функции полезности доказывается, что функция f , удовлетворяющая условиям (1), может быть определена на бесконечном множестве в том и только в том случае, если это множество содержит счетное, всюду плотное подмножество (является сепарабельным). В данной статье исследуются лишь задачи с конечным множеством решений, для которых существование функции полезности очевидно.

Таким образом, необходимо построение числовой функции полезности на основе имеющейся информации о предпочтениях. Эта информация может быть формализована различным образом в зависимости от возможности ЛПР при сравнении решений [1]. Если отношение предпочтения на множестве альтернатив определить затруднительно, но можно оценить вероятность $p(x, y)$ того, что альтернатива x предпочтительнее альтернативы y из X [2], то для выявления функции полезности можно положить

$$\begin{aligned} x \succ y, & \text{ если } p(x, y) > \frac{1}{2}; \\ x \sim y, & \text{ если } p(x, y) = \frac{1}{2}. \end{aligned} \quad (2)$$

В случае невозможности оценить вероятность $p(x, y)$ целесообразно сравнить различные вероятности $p(x, y)$ и $p(u, v)$ для всех $x, y, u, v \in X$, что позволит получить информацию вида $p(x, y) > p(u, v)$.

Для построения функции полезности ответы ЛПР должны удовлетворять следующим условиям:

1. $p(x, y) + p(y, x) = 1$ (3), которое обеспечивает связность устанавливаемого отношения.

2. $p(x, y) < p(u, v) \rightarrow [f(x) - f(y)] < [f(u) - f(v)]$ (4), которое устанавливает зависимость между вероятностью предпочтения альтернатив и разностью их полезности и может быть эквивалентно записано в виде $p(x, y) < p(u, v) \rightarrow p(x, u) < p(y, v)$ (5).

В случае, когда предпочтения заданы в виде вероятностей, в работе Дебре [3] доказывается следующая теорема существования функции полезности.

Теорема Дебре. Пусть на множестве X задано бинарное отношение предпочтения (2). Для вероятностей выполняются соотношения (3), (5), а также некоторое условие непрерывности (аналог аксиомы Архимеда):

для любых x, y, z из X и числа q таких, что $p(x, z) < q < p(y, z)$, существует такое решение $v \in X$, что $p(v, z) = q$. Тогда существует функция полезности, для которой выполняется соотношение (4), единственная с точностью до возрастающего линейного преобразования [1].

В работе [4] рассматривается аналогичная задача, названная проблемой упорядочения разностей. Решения при этом сравниваются между собой парами, т. е. фактически на множестве X установлено четырехместное отношение предпочтения — бинарное отношение предпочтения на множестве $X \times X$. Теорема, устанавливающая необходимые и достаточные условия для существования функции полезности, формулируется следующим образом.

Теорема Скотт. Пусть X — непустое конечное множество решений, $\succsim$ — четырехместное отношение предпочтения на нем.

Для существования функции полезности такой, что $(x, y) \succeq (u, v) \Leftrightarrow [f(x) - f(y)] > [f(u) - f(v)]$ (6), необходимо и достаточно, чтобы выполнялись следующие условия для всех x, y, u, v из X : 1) выполняется либо $(x, y) \succeq (u, v)$, либо $(u, v) \succeq (x, y)$; 2) если $(x, y) \succeq (u, v)$, то $(v, u) \succeq (y, x)$; 3) для любых последовательностей $x_0, \dots, x_{n-1}, y_0, \dots, y_{n-1}$ из X и любых перестановок π и σ из множества индексов $\{0, \dots, n-1\}$ из $(x_i, y_i) \succeq (x_{\pi(i)}, y_{\sigma(i)})$ для всех $n > 0$ и $0 < i < n$ следует, что $(x_{\pi(0)}, y_{\sigma(0)}) \succeq (x_0, y_0)$.

Условие 3) теоремы, которое нужно для транзитивности устанавливаемого отношения предпочтения, представляет собой бесконечную систему условий — по одному для каждого $n > 0$ и перестановок индексов π и σ [4]. В работе [5] Скотт и Сапс доказали, что не существует никакой конечной системы таких условий, достаточных для существования функций полезности (6). Для решения практических задач проверка этих условий, безусловно, не представляется возможной.

Итак, желательно найти более простые ограничения, накладываемые на модель предпочтения ЛПР, чтобы получить на множестве возможных решений транзитивное отношение предпочтения для существования наилучшего решения и определить на нем функцию полезности.

Рассмотрим задачу принятия решений в многокритериальной постановке, когда предпочтение определяется на основе оценок решений по нескольким критериям эффективности. В работе Тверского [6] для решений таких задач представлены алгоритмы, известные как «методы сравнения разностей оценок альтернатив». В частности, предлагается следующая процедура сравнения решений по предпочтению. Пусть $x, y \in X$, $(K_1, \dots, K_N)$ — множество критериев эффективности, $K_s: X \rightarrow R^1$, $s = \overline{1, N}$, составляющих в совокупности векторный критерий эффективности. Отношение предпочтения определяется следующим образом: $x \succeq y$, если имеет место неравенство

$$\sum_{s=1}^N \varphi_s [K_s(x) - K_s(y)] \geq 0, \quad (7)$$

где φ_s — функция, определяющая влияние разностей оценок по s -му критерию на результат сравнения двух решений. Неравенство (7) задает группу методов. Рассмотрим условия, при которых гарантируется транзитивность устанавливаемого отношения предпочтения на множестве X : 1) при $N = 1$ (один критерий) это правило, очевидно, всегда транзитивно; 2) при $N = 2$ данное правило транзитивно тогда и только тогда, когда $\varphi_1(x) = \varphi_2(t \cdot x)$, где t — положительная константа; 3) при $N > 3$, как это и бывает в большинстве реальных многокритериальных задач, правило (7) дает транзитивное отношение предпочтения

в том и только в том случае, когда все φ_s ($s = \overline{1, N}$) линейны, т. е. $\varphi_s(x) = t_s x$, где t — положительная константа.

Ввиду жесткости этих условий важной задачей является исследование транзитивности правила сравнения решений типа (7).

Предлагается методика формирования многокритериальной функции полезности на множестве допустимых решений, теоретической основой которой являются результаты, изложенные авторами в работах [7, 8].

Введем следующие обозначения:

$$\sum_{s=1}^N \varphi_s [K_s(x) - K_s(y)] = \Phi(x, y), \quad (8)$$

где x, y — решения из X . В соответствии с введенными понятиями функцию $\Phi(x, y)$ рассмотрим как степень превосходства (СП) решения x над решением y . Отношение предпочтения на множестве решений определим так:

$$R_l = \{(x, y) | \Phi(x, y) \geq l; x, y \in X\}. \quad (9)$$

Естественно брать величину уровня $l = [0, \max_{x, y \in X} \Phi(x, y)]$. Вели-

чина эта выбирается в зависимости от характера задачи. Заметим, что правило (7) является частным случаем правила (9) при $l = 0 (R_0)$.

В соответствии с [7, 8] справедливы утверждения.

Утверждение 1. Если СП $\Phi(x, y)$ удовлетворяет свойству $\Phi(x, y) = \Phi(x, z) + \Phi(z, y)$ для всех $x, y, z \in X$ (10), то R_l транзитивно для всех разрешенных значений. Это утверждение доказывается непосредственно из определения отношения предпочтения (9).

Утверждение 2. Если $\Phi(x, y) = f(x) - f(y)$, то функция $\Phi(x, y)$ является кососимметричной: $\Phi(x, y) = -\Phi(y, x)$ (11), а также для нее выполняется условие (10). Справедливо и обратное утверждение.

Утверждение 3. Если $\Phi(x, y)$ — кососимметричная функция и обладает свойством (10), то $\Phi(x, y) = f(x) - f(y)$, при этом

$$f(x) = \sum_{z \in X} \lambda(z) \Phi(x, z), \quad (12)$$

где здесь и в дальнейшем $\lambda(z) \geq 0$ для всех $z \in X$ и $\sum_{z \in X} \lambda(z) =$

$= 1$. Функция $f(\cdot)$ выступает в качестве функции полезности, определенной на множестве X . В [7, 8] доказывается, что эта функция измерена в шкале интервалов. Так как функция $\Phi(x, y)$ — степень превосходства — построена на основе многокритериальной модели, то f может интерпретироваться как многокритериальная функция полезности. Функцию $\Phi(x, y)$ можно вычислять

различными способами, не нарушая ее основных свойств. В частности, разработано четыре способа построения ее на основе многокритериальной модели принятия решений [7, 8]. По своей математической постановке адекватно могут быть представлены три различных типа задач принятия решений: многокритериальные, группового выбора и принятия решений при неопределенности и риске. Поэтому полученные результаты могут быть обобщены и для этих классов задач.

Результат работы [6] для случая $N \geq 3$ является частным случаем полученных результатов, а именно: если все φ_s линейны в формуле (7), т. е. имеет место

$$\varphi_s(x, y) = t_s [K_s(x) - K_s(y)],$$

то на основе утверждения 2 все φ_s кососимметричны и обладают свойством (10). Докажем, что тогда этими же свойствами обладает и их сумма.

Утверждение 4. 1. Если $\varphi_s(x, y) = -\varphi_s(y, x)$ для всех $s = \overline{1, N}$, то $\Phi(x, y)$, определенная по формуле (6), тоже обладает этим свойством. 2. Если $\varphi_s(x, y) = \varphi_s(x, z) + \varphi_s(z, y)$ для всех $x, y, z \in X$, то и $\Phi(x, y)$ обладает этим же свойством.

Доказательство. Покажем сначала справедливость первой части утверждения:

$$\Phi(y, x) = \sum_{s=1}^N \varphi_s(y, x) = - \sum_{s=1}^N \varphi_s(x, y) = -\Phi(x, y),$$

что и требовалось доказать.

Вторая часть утверждения тоже доказывается путем несложных преобразований:

$$\begin{aligned} \Phi(x, y) &= \sum_{s=1}^N \varphi_s(x, y) = \sum_{s=1}^N [\varphi_s(x, z) + \varphi_s(z, y)] = \\ &= \sum_{s=1}^N \varphi_s(x, z) + \sum_{s=1}^N \varphi_s(z, y) = \Phi(x, z) + \Phi(z, y). \end{aligned}$$

Ввиду утверждения 1, правило сравнения решений, представленное формулой (7), транзитивно в общем случае ($N \geq 3$).

Сравним предложенный алгоритм с моделью стохастического предпочтения, рассмотренной в [1]. Между СП $\Phi(x, y)$ и $p(x, y)$ из формулы (3) получается следующая взаимосвязь:

$$\left(p(x, y) - \frac{1}{2}\right) = -\left(p(y, x) - \frac{1}{2}\right)$$

и можно положить $\Phi(x, y) = p(x, y) - 1/2$ (13), т. е. условие 1 теоремы Дебре для СП проявляется в ее кососимметричности.

Учитывая выражение (13), при условии, что для СП выполняется соотношение (10), легко можно вывести и второе условие этой теоремы для $\Phi(x, y)$: $\Phi(x, y) < \Phi(y, u) \Leftrightarrow \Phi(x, u) < \Phi(y, u)$.

Условие 3 теоремы Дебре не имеет смысла и не требуется для конечных множеств решений X . Поэтому можно заключить, что теорема Дебре справедлива для модели со степенью превосходства. Более того, утверждение 3 дает конкретный вид этой функции полезности (12).

Условие (10) является довольно жестким, однако выполнение его, очевидно, очень желательно. На основе СП строится функция, называемая интегральной степенью превосходства (ИСП):

$$\tilde{\Phi}(x, y) = \sum_{z \in X} \lambda(z) [\Phi(x, z) - \Phi(y, z)]. \quad (14)$$

ИСП обладает следующими свойствами: а) кососимметрична — $\tilde{\Phi}(x, y) = -\tilde{\Phi}(y, x)$; б) $\tilde{\Phi}(x, y)$ всегда обладает свойством (10); в) $\tilde{\Phi}(x, y) = \tilde{\Phi}(x, z) + \tilde{\Phi}(z, y)$; г) ИСП совпадает с СП ($\tilde{\Phi}(x, y) = \Phi(x, y)$), когда СП в свою очередь удовлетворяет свойствам (9) и (10).

На основе утверждения 3 и свойств ИСП получаем зависимость $\tilde{\Phi}(x, y) = \tilde{f}(x) - \tilde{f}(y)$, где $\tilde{f}(x) = \sum_{z \in X} \lambda(z) \tilde{\Phi}(x, z)$ (15). Функ-

ция $\tilde{f}(\cdot)$ тоже является многокритериальной функцией полезности в указанном выше смысле. Нетрудно доказать справедливость следующего утверждения.

Утверждение 5. Если $\Phi(x, y)$ — кососимметрична, то $\tilde{f}(x) = f(x)$, где $\tilde{f}(x)$ и $f(x)$ определены по формулам (15) и (12) соответственно.

Отношение предпочтения на множестве решений определим на основе ИСП аналогичным образом:

$$\tilde{R}_I = \{(x, y) | \tilde{\Phi}(x, y) \geq 1, x, y \in X\}. \quad (16)$$

На основе утверждения 1 и свойств ИСП отношение предпочтения $\tilde{R}_I$ всегда транзитивно для всех разрешенных значений I .

Итак, с помощью формулы (14) мы получаем функцию, которая описывает предпочтение решения x над решением y и обладает при этом нужными свойствами (10) и (11). С ее помощью и строится транзитивное бинарное отношение предпочтения на X и функция полезности, заданная в шкале интервалов, т. е. единственная с точностью до возрастающего линейного преобразования.

Если степень превосходства вычислять по формуле (13), то отношение R_I можно определить для модели со стохастическим предпочтением. Перейдя затем к интегральной степени превосходства по формуле (14), мы получим всегда транзитивное отношение предпочтения $\tilde{R}_I$ и функцию полезности, не требуя

при этом выполнения довольно искусственного условия 2 теоремы Дебре.

Результаты, изложенные в данной статье, используются в диалоговой процедуре для выбора наилучшего решения на основе выявления системы предпочтения ЛПР или экспертов путем попарного сравнения возможных решений [9]. Программная реализация диалогового алгоритма реализована на алгоритмическом языке ПЛ-1 в среде ОС ЕС.

Список литературы: 1. Вилкас Э. Я., Майминс Е. Э. Решения: теория, информация, моделирование. — М.: Радио и связь, 1981. — 328 с. 2. Жуковин В. Е. Модели и процедуры принятия решений. — Тбилиси: Мецниереба, 1981. — 111 с. 3. Debreu G. Stochastic choice and cardinal utility. — *Econometrica*, 1958, 26, p. 440—444. 4. Scott D. Measurement structures and linear inequalities. — *Journ. Math. Psychology*, 1964, 1, p. 233—247. 5. Scott D. and Suppes P. Foundational aspects of theories of measurement. — *Journ. Symbolic Logic*, 1958, 23, p. 113—128. 6. Tversky A. Intransitivity of preferences. — *Psychol Rev.*, 1969, 76 № 1. 7. Жуковин В. Е. К проблеме группового выбора. — *Сообщения АН ГССР*, 1980, 99, № 2, с. 333—336. 8. Буриштейн Ф. В. Об одном подходе к многокритериальным задачам принятия решений. — *Сообщения АН ГССР*, 1980, 100, № 3, с. 569—572. 9. Жуковин В. Е., Буриштейн Ф. В., Заславский А. Б. Диалоговая процедура выбора наилучшего решения по многим критериям эффективности при неполной информации. — В кн.: Взаимодействие человека и ЭВМ в процессе принятия решений по многим критериям эффективности. — Кутавси, 1980, с. 66—68.

Поступила в редколлегию 03.02.82.

УДК 658.012

А. В. ФИЛИМОНОВ

ПРИМЕНЕНИЕ КОНЦЕПЦИИ РАСШИРЕННОЙ МОДЕЛИ МНОГОСВЯЗНЫХ ОБЪЕКТОВ К ЗАДАЧАМ АВТОНОМНОГО РЕГУЛИРОВАНИЯ

Концентрация расширенной модели конечномерных объектов. Положим, что динамика объекта n -го порядка с r -мерным управляющим входом описывается уравнением

$$\frac{d}{dt} x(t) = Ax(t) + Bu(t), \quad (1)$$

где $t \in R_+$, $x \in R^n$ — состояние объекта; $u = \text{col}(u_1, \dots, u_r) \in K^r$ — векторный вход, а u_k , $k = 1, \dots, r$, — его скалярные компоненты; $A \in R^{n \times n}$ и $B \in R^{n \times r}$ — коэффициентные матрицы объекта. Здесь и далее R — ось вещественных чисел; R_+ — ее положительная полуось.

Выделим в матрице B вектор-столбцы B_k , $k = 1, \dots, r$: $B = [B_1, \dots, B_r]$. Представим передаточную матрицу объекта в виде

$$W(p) = (I_p - A)^{-1}B = \frac{M(p)}{\Delta(p)},$$

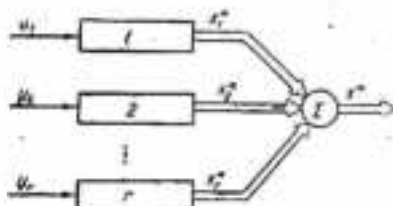
где p — комплексная частота; $\Delta(p)$ — характеристический полином, а $M(p)$ — полиномиальная матрица размера $n \times r$:

$$\Delta(p) = \det(I_p - A) = p^n + \sum_{i=0}^{n-1} \alpha_i p^i, \quad M(p) = \sum_{i=0}^{n-1} M_i p^i.$$

Здесь I — единичная матрица n -го порядка, $\alpha_i \in R$ и $M_i \in R^{n \times r}$, $i = 0, \dots, n-1$. Выделим в матрицах M_i вектор-столбцы M_{ik} : $M_i = [M_{i1}, \dots, M_{ir}]$.

Рассмотрим динамическую систему $N = n \times r$ -го порядка, функционирующую согласно уравнениям

$$\frac{d^n}{dt^n} \xi_k(t) + \sum_{i=0}^{n-1} \alpha_i \frac{d^i}{dt^i} \xi_k(t) = u_k(t), \quad k = 1, \dots, r. \quad (2)$$



$$x_k^M(t) = \sum_{i=0}^{n-1} M_{ik} \frac{d^i}{dt^i} \xi_k(t); \quad (3)$$

$$x^M(t) = \sum_{k=1}^r x_k^M(t), \quad (4)$$

$t \in R_+$, $\xi_k \in R$, $x_k^M \in R^n$, $k = 1, \dots, r$ и $x^M \in R^n$ — выход системы.

Рис. 1. Структура расширенной модели объекта с r параллельно работающими однотипными подсистемами

определяющая отклик ее выхода на вход, совпадает с передаточной матрицей объекта $W(p)$. Отсюда следует, что при полностью управляемом объекте любое его движение $x(t)$ можно воспроизвести на выходе рассматриваемой системы, т. е., если ее привести в надлежащее состояние, то получим $x^M(t) = x(t)$ (5).

Таким образом, динамическая система (2) + (4) моделирует [1] объект (1). Если $r > 1$, то $N > n$, так что данную систему будем называть *расширенной моделью* объекта.

Структуру расширенной модели отражает рис. 1: динамика модели сосредоточена в r параллельно работающих однотипных подсистемах с одномерными входами $u_1, \dots, u_r$, описываемых уравнениями (2) и (3). Эти подсистемы будем называть *сепаратными*, имея в виду их функциональную независимость. Текущее состояние k -й сепаратной подсистемы задается n -мерным вектором $z_k(t) = \text{col}(\xi_k(t), \dots, \xi_k^{(n-1)}(t))$ (6).

Для систем с такой структурой чрезвычайно упрощается задача регулирования, ибо она декомпозируется на r более простых задач односвязного регулирования движения сепаратных подсистем: так, в K -й сепаратной подсистеме скалярное управление $U_k(t)$ направлено исключительно на переменную $\xi_k(t)$.

Ясно, что задачу программного управления исходным объектом можно переформулировать на его расширенную модель

и затем на r более простых задач управления сепаратными подсистемами.

Автономное регулирование. Концепцию расширенной модели можно применять для синтеза систем автономного регулирования, отправная идея при этом — расширение пространства состояния объекта путем наращивания его структуры. Суть идеи состоит в том, чтобы подключить к объекту специальное динамическое звено $(N - n)$ -го порядка, которое назовем *корректирующим*, с тем, чтобы скорректированный объект (исходный объект + корректирующее звено) стал эквивалентным расширенной модели исходного объекта. В этом случае можно будет считать, что уравнения (2) — (4) представляют динамику скорректированного объекта. Наметим общую схему реализации этой идеи, не углубляясь в ее алгоритмические аспекты.

Допустим, что объект полностью управляем. Тогда его расширенную модель можно расчленить на две подсистемы: полностью наблюдаемую и полностью ненаблюдаемую по выходу модели x^M . Первая, согласно (5), имеет n -й порядок, эквивалентна объекту и может быть представлена уравнением, идентичным (1):

$$\frac{d}{dt} x^M(t) = Ax^M(t) + Bu(t).$$

Вторая подсистема — порядка $n_1 = N - n$ испытывает воздействие первой, и динамика ее состояния $x_1 \in R^{n_1}$ подчиняется уравнению вида

$$\frac{d}{dt} x_1^M(t) = A_1 x_1^M(t) + B_1 u(t) + D x^M(t), \quad (7)$$

где $A_1 \in R^{n_1 \times n_1}$, $B_1 \in R^{n_1 \times r}$ и $D \in R^{n_1 \times n}$.

Переменные z_k должны линейно выражаться через координаты состояния модели: $z_k(t) = L_k x^M(t) + L_k^1 x_1^M(t)$, где $L_k \in R^{n \times n}$, $L_k^1 \in R^{n \times n_1}$, $k = 1, \dots, r$.

Теперь становится ясно, каким путем наращивать структуру объекта с тем, чтобы добиться его эквивалентности расширенной модели. Для этого следует подключить к объекту корректирующее звено, функционирующее согласно уравнениям

$$\frac{d}{dt} x_1(t) = A_1 x_1(t) + B_1 u(t) + D x(t); \quad z_k(t) = L_k x(t) + L_k^1 x_1(t), \\ k = 1, \dots, r.$$

Тогда динамику скорректированного объекта в переменных $Z_1, \dots, Z_r$ опишем уравнениями (2) (см. рис. 2): коррекция объекта (рис. 2, а) равносильна его расчленению на r сепаратных подсистем со скалярным управляющим входом.

Таким образом, исходная задача многосвязного регулирования распадается на r задач односвязного регулирования. Действительно, состояние (6) k -й сепаратной подсистемы (2) регулируется исключительно компонентой u_k управляющего входа объекта. При этом регулируемое движение данной подсистемы

не зависит от регулируемого движения остальных сепаратных подсистем. В итоге система регулирования включает в свою структуру корректирующее звено, развязывающее каналы управления, и r однотипных регуляторов, функционирующих автономно и замыкающих соответствующие сепаратные подсистемы.

Изложенные результаты в идейном плане перекликаются с результатом М. Уонэма [2] по автономному управлению многомерными объектами.

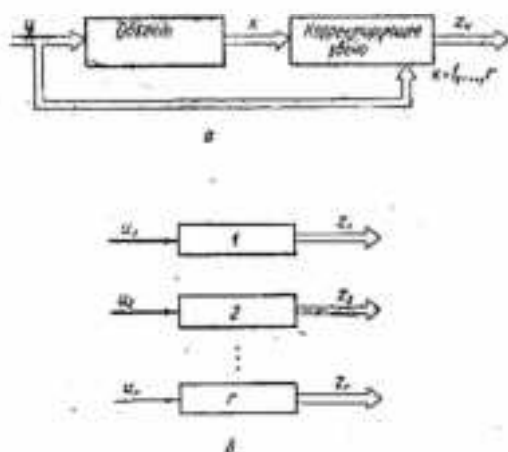


Рис. 2. Структурная модель скорректированного объекта

Подчеркнем также следующую деталь: построенное корректирующее звено, подключаемое каскадно к объекту, как показано на рис. 2, а, в скрытом виде заключает в себе модель динамики объекта, ибо его характеристические числа — собственные значения матрицы A_1 в (7) — совпадают с характеристическими числами объекта. В этом трудно убедиться, если принять во внимание, что спектр характеристических чисел расши-

ренной модели представляет собою r -кратно взятый спектр моделируемого объекта.

Модельное управление. Описанная динамическая развязка каналов управления позволяет весьма просто решать задачу модального управления объектом.

При замыкании каждой сепаратной подсистемы собственным регулятором, функционально не связанным с другими, система регулирования распадается на r параллельно работающих контуров регулирования. Следовательно, каждому из этих контуров можно навязать желаемый спектр характеристических чисел, в результате чего спектр Λ замкнутой системы включает в себя спектры Λ_k отдельных контуров регулирования $\Lambda \supset \Lambda_k, k = 1, \dots, r$.

Положим, что желаемый характеристический полином k -й сепаратной подсистемы равен

$$\Delta_k(p) = p^n + \sum_{i=0}^{n-1} \beta_i^k p^i.$$

Тогда модальный регулятор для k -й сепаратной подсистемы должен формировать регулирующее воздействие по закону $u_k(t) = H_k z_k(t)$,

где

$$H_k = [\alpha_0 - \beta_0^k, \alpha_1 - \beta_1^k, \dots, \alpha_{n-1} - \beta_{n-1}^k].$$

Посредством r автономно работающих модальных регуляторов навязем контурам регулирования характеристические полиномы $\Delta_1(p), \dots, \Delta_r(p)$. В итоге характеристический полином замкнутой системы будет иметь вид $\delta(p) = \prod_{k=1}^r \Delta_k(p)$.

Положим, что желаемый модальный состав регулируемого движения $x(t)$ объекта задается характеристическим полиномом $\Delta_k(p)$. Тогда следует положить $\Delta_k(p) = \Delta_k(p)$, $k = 1, \dots, r$.

Недостаток предложенного решения задачи модального управления по сравнению с традиционными [3] состоит в том, что оно приводит к увеличению динамического порядка системы. Однако, бесспорно, схемное достоинство решения — простота перенастройки системы на любой заданный спектр характеристических чисел, не требующей изменения структуры системы.

Редукция модели. Нетрудно понять, что описанная модель объекта, вообще говоря, имеет избыточную структуру и ее можно редуцировать. Покажем, каким путем удастся построить модель объекта меньшего порядка с тем же целевым назначением.

Реакция объекта на k -ю компоненту управления определяется передаточной матрицей размера $n \times 1$: $W_k(p) = (I_p - A)^{-1} B_k$ (8).

Положим, что вполне управляемая входной переменной u_k подсистема объекта имеет динамический порядок n_k , а ее характеристический полином равен $\Delta_k(p)$:

$$\Delta_k(p) = p^{n_k} + \sum_{i=0}^{n_k-1} \alpha_i^k p^i.$$

Тогда передаточная матрица (8) имеет следующий вид:

$$W_k(p) = \frac{M_k(p)}{\Delta_k(p)} = \frac{1}{\Delta_k(p)} \sum_{i=0}^{n_k-1} M_{ki} p^i, \quad (9)$$

где $M_k(p)$ — полиномиальная, а M_{ki} , $i = 0, \dots, n_k - 1$, — числовые матрицы размера $n \times 1$.

Отметим, что n_k можно определить как ранг матрицы управляемости пары (A, B_k) : $n_k = \text{rang}[B_k, AB_k, \dots, A^{n-1}B_k]$ (10).

Передаточная матрица $W(p)$ объекта равна

$$W(p) = (I_p - A)^{-1} B = [W_1(p), \dots, W_r(p)]. \quad (11)$$

Рассмотрим динамическую систему N -го порядка $N = \sum_{k=1}^r n_k$,

образованную параллельным соединением r динамических подсистем, описывающихся уравнениями

$$\Delta_k \left(\frac{d}{dt} \right) \xi_k(t) = \frac{d^{n_k}}{dt^{n_k}} \xi_k(t) + \sum_{i=0}^{n_k-1} \alpha_i^k \frac{d^i}{dt^i} \xi_k(t) = u_k(t); \quad (12)$$

$$x_k^M(t) = M_k \left(\frac{d}{dt} \right) \xi_k(t) = \sum_{i=0}^{n_k-1} M_{ki} \frac{d^i}{dt^i} \xi_k(t), \quad (13)$$

где $t \in R_+$, $\xi_k \in R$ — скалярная внутренняя переменная k -й подсистемы, $x_k^M \in R^n$ — ее векторный выход, так что вектор выходных переменных системы $x^M \in R^n$ определяется уравнением

$$x^M(t) = \sum_k x_k^M(t). \quad (14)$$

Текущее состояние k -й сепаратной подсистемы определяется на этот раз n_k -мерным вектором

$$z_k(t) = \text{col}(\xi_k(t), \dots, \xi_k^{(n_k-1)}(t)).$$

Итак, рассматриваемая динамическая система образована параллельным соединением r сепаратных подсистем с одномерным управляющим входом. Кроме того, ее передаточная функция совпадает с передаточной функцией (11) объекта. Следовательно, эту систему можно применять как модель объекта по аналогии с ранее рассмотренной моделью (2) — (4).

При $r > 1$ динамический порядок объекта меньше порядка рассматриваемой модели: $n > N$, так что последнюю можно также называть расширенной моделью объекта.

Анализ предельных по быстрдействию процессов регулирования. Концепция расширенной модели позволяет уточнить известные представления об оптимальных по быстрдействию процессах регулирования в многосвязных объектах.

Пусть задана ограниченная область допустимых значений управления $U \subset R^r$: $u(t) \in U$. Положим, что U — r -мерный параллелепипед, задаваемый неравенствами $u_k^{\min} < u_k < u_k^{\max}$ (15), где $-\infty < u_k^{\min} < 0 < u_k^{\max} < +\infty$, $k = 1, \dots, r$.

Наряду с объектом будем анализировать движение его расширенной модели, описываемой управлениями (12) — (14).

Положим, что в начальный момент времени $t = 0$ объект находится в некотором состоянии $x(0)$, а $u_{\text{opt}}(t)$ — оптимальное управляющее воздействие, переводящее объект из этого состояния в положение равновесия $x = 0$ за время T . Покажем, что начальному состоянию $x(0)$ объекта можно поставить в соответствие начальное состояние $z(0)$ модели, по отношению к которому управление $u_{\text{opt}}(t)$ также является оптимальным. С этой целью рассмотрим управление $u_{\text{opt}}^*(t)$, получающееся временным обращением сигнала $u_{\text{opt}}(t)$: $u_{\text{opt}}^*(t) = u_{\text{opt}}(T - t)$ при $0 \leq t \leq T$. Нетрудно понять, что первоначально покоящаяся модель под действием входного воздействия к моменту времени $t = T$ перейдет в требуемое состояние $z^*(T) = z(0)$.

Итак, если программа управления $u_{\text{opt}}(t)$ оптимальна для некоего начального состояния объекта, то она будет также оп-

тимальной и для некоторого состояния модели, т. е. структуру оптимальных регулирующих воздействий объекта можно выяснить посредством изучения процессов регулирования в модели.

Обращаясь к k -й сепаратной подсистеме (см. уравнение (12)), можно сделать вывод, что оптимальная управляющая программа $u_{\text{opt}}(t)$ определяется исключительно начальным состоянием $z_k(0)$ этой подсистемы. Но тогда, согласно известной теореме А. А. Фельдбаума об $n_k - 1$ переключениях [4], управление $u_{\text{opt}}(t)$ является релейной функцией, принимающей граничные значения отрезков (15) в течение времени успокоения T_k подсистемы с не более чем $n_k - 1$ переключениями. Ясно, что если T — время успокоения модели, то на интервале (T_k, T) можно положить $u_{\text{opt}}(t) = 0$.

Итак, мы пришли к следующей теореме, которая распространяет теорему А. А. Фельдбаума на класс многосвязных по управлению объектов и является более тонким результатом, чем [5, с. 64].

Теорема. Если пара (A, B) вполне управляема, то для произвольного начального состояния объекта найдется такое оптимальное по быстродействию управление $u_{\text{opt}}(t)$, каждая компонента которого $u_{k\text{opt}}(t)$ является до некоторого конечного момента времени релейной функцией времени t с не более чем $n_k - 1$ переключениями, где n_k определяется соотношением (10), принимающей предельные значения $u_k^{\min}$ и $u_k^{\max}$, после чего она тождественно равна нулю.

Список литературы: 1. Калман Р., Фалб П., Арbib М. Очерки по математической теории систем.—М.: Мир, 1971.—400 с. 2. Уокен М. Линейные многомерные системы управления. Геометрический подход.—М.: Наука, 1980.—376 с. 3. Porter B., Grossley R. Modal Control; Theory and Applications.—London: Taylor E. Francis, 1972. 4. Фельдбаум А. А., Бутковский А. Г. Методы теории автоматического управления.—М.: Наука, 1971.—743 с. 5. Лейтман Дж. Введение в теорию оптимального управления.—М.: Наука. 1968.—190 с.

Поступила в редакцию 29.12.81.

УДК 658.012.011.56

А. Ф. ЗОЗУЛЯ, П. Г. РОМАНИЙ, канд. техн. наук,
Г. Г. АСЕЕВ

ПОСЛЕДОВАТЕЛЬНО-ДЕКОМПОЗИЦИОННЫЙ МЕТОД
РАСЧЕТА БАЛАНСОВЫХ УРАВНЕНИЙ СЛОЖНЫХ
ТЕХНОЛОГИЧЕСКИХ КОМПЛЕКСОВ С ОБРАТНЫМИ
ПОТОКАМИ

Главным этапом технологического проектирования химических производств является расчет балансовых уравнений, по результатам которого принимаются основные проектные решения. Современные химико-технологические системы (ХТС) ха-

рактизуются большим числом взаимосвязанных аппаратов, значительным количеством потоков, наличием многочисленных технологических ограничений на режимные параметры, а также нелинейным характером зависимостей, описывающих отдельные процессы. Большая размерность системы балансовых уравнений ХТС и наличие обратных потоков в системе требуют разработки специальных методов расчета материальных балансов, учитывающих специфику данной задачи.

Традиционный подход при расчете ХТС с обратными потоками [1, 2] состоит в выявлении свободных элементов и групп элементов (блоков), охваченных обратными потоками. Выделенные свободные элементы и блоки образуют разомкнутую систему, которая рассчитывается затем в определенной последовательности. Для расчета блоков, представляющих собой в общем случае многоконтурные системы, используется метод разрыва технологических связей, а математически расчет каждого контура методом разрыва сводится к определению параметров разрываемых потоков на основе решения системы нелинейных уравнений. Недостатками такого подхода являются необходимость использования для выделения блоков достаточно сложных алгоритмов структурного анализа и трудности вычислительного характера, связанные с оценкой сходимости наиболее часто используемых при решении систем нелинейных уравнений методов простой итерации, Ньютона-Рафсона и Вольфа во всей допустимой области варьирования переменных.

В работах [3, 4] изложен без строгого обоснования метод расчета сложных ХТС с обратными потоками, не требующий выделения групп элементов системы, охваченных обратными потоками, и экспериментально подтверждена его работоспособность. В настоящей статье проведен анализ структуры вычислительной процедуры предлагаемого последовательно-декомпозиционного метода, исследована его сходимость применительно к задаче расчета балансовых уравнений ХТС, приведены рекомендации по решению отдельных подзадач, образующих вычислительную последовательность.

Уравнения материальных балансов (МБ) множества элементов ХТС имеют следующий вид: — материальный баланс по общему массовому расходу вещества

$$\sum_{i \in I_p^+} m_i - \sum_{i \in I_p^-} m_i = 0, \quad p \in \Omega \quad (1), \quad \text{где}$$

I_p^+ , I_p^- — множества индексов соответственно входных и выходных потоков p -го элемента; m_i — массовый расход i -го потока; — материальный баланс по массовому расходу

$$\sum_{i \in I_p^+} m_i C_{ij} - \sum_{i \in I_p^-} m_i C_{ij} = 0, \quad j = \overline{1, k}, \quad p \in \Omega, \quad (2)$$

где C_{ij} — доля j -го компонента в i -м потоке; k — число компонент в потоке.

Обозначим через X^0 область допустимых значений вектора переменных уравнений МБ $\bar{x} = (x_1, x_2, \dots, x_n)$, которая имеет следующий вид: $X^0 = \{(x_1, x_2, \dots, x_n) | x_i^{\min} \leq x_i \leq x_i^{\max}, i = \overline{1, n}; \bar{x}^{(p)} - \bar{x}^{(l)} = 0, (p, l) \in I\}$, где $x_i^{\min}, x_i^{\max}$ — соответственно минимальное и максимальное значения i -й переменной; $n = \dim \bar{x}$; I — множество пар элементов ХТС, связанных общими потоками; $\bar{x}^{(p)}, \bar{x}^{(l)}$ — векторы смежных переменных p -го и l -го элементов; $p, l \in \Omega$.

Вектор невязок уравнений МБ p -го элемента ХТС обозначим $\bar{f}^{(p)} = [f_1^{(p)}, f_2^{(p)}, \dots, f_{k_p}^{(p)}]$, $p \in \Omega$. Тогда функцию невязки всей системы уравнений МБ можно записать так:

$$F = \frac{1}{2} \sum F^{(p)} = \frac{1}{2} \sum_{p \in \Omega} \sum_{i=1}^{k_p} (f_i^{(p)})^2. \quad (3)$$

При введенных обозначениях задача расчета балансовых уравнений сводится к нахождению $\min \{F(\bar{x}) | \bar{x} \in X^0\}$. Представим ХТС в виде ориентированного циклического потокового графа, вершины — стоки и вершины, источники которого объединены в один 0-узел. В дальнейшем невязку на 0-узле будем обозначать $F^{(0)}$.

Обозначим через $n_0, n_1, \dots, n_N$ размерности векторов переменных соответственно 0-узла, первого элемента, N -элемента циклического потокового графа ХТС, где под элементами будем понимать отдельные узлы и группы узлов графа. Области допустимых значений векторов переменных элементов в вещественных пространствах $Rn_0, Rn_1, \dots, Rn_N$ обозначим соответственно $X_0, X_1, \dots, X_N$. Образует множества: $\bar{X}_i = X_i \setminus (X_{i-1} \cap X_i) i = \overline{1, N}$, $\bar{X}_0 = X_0 \setminus (X_N \cap X_0)$. Процедура минимизации функции F_2 , равной

$$F_2 = F + F^{(0)} = \frac{1}{2} \sum_{j=1}^{nN} \sum_{i=1}^{k_j} (f_i^{(j)})^2,$$

где k_j — число уравнений балансов j -го элемента ХТС, с помощью предлагаемого последовательного метода выглядит следующим образом: выбирается некоторый допустимый вектор переменных $\bar{x}^0 = [\bar{x}_0(0), \bar{x}_1(0), \dots, \bar{x}_{nN}(0)]$, где $\bar{x}_i \in \bar{X}_i, i = \overline{0, nN}$ и для его улучшения решается задача $\min F_2(\bar{x}_0, \bar{x}_1, \dots, \bar{x}_{nN})$ (4) при $\bar{x} \in \bar{X}_0$ и фиксированных значениях $\bar{x}_i(0), i = \overline{1, nN}$.

В результате получаем новое решение $\bar{x}^{1/nN} = [\bar{x}(1), \bar{x}_1(0), \dots, \bar{x}_{nN}(0)]$. Затем при фиксированных значениях $\bar{x}_0(1), \bar{x}_2(0), \dots, \bar{x}_{nN}(0)$ задача (4) решается путем варьирования $\bar{x}_1 \in \bar{X}_1$. Полу-

чаем новый вектор переменных: $\tilde{x}^{2/N}(\tilde{x}_0(1), \tilde{x}_1(1), \dots, \tilde{x}_{n_N}(0))$ и так далее. После $(N_n + 1)$ шагов получаем $\tilde{x}^1 = (\tilde{x}(1), \tilde{x}_1(1), \dots, \tilde{x}_{n_N}(1))$ и первый итерационный шаг по всей схеме заканчивается. Пусть выполнено k итераций по схеме. Тогда $(k + 1)$ итерация заключается в отыскании улучшенных векторов $\tilde{x}_p(k + 1)$, $p = \overline{0, N_n}$ путем минимизации функции $F_x(\tilde{x}_0(k + 1), \dots, \tilde{x}_{p-1}(k + 1), \tilde{x}_p(k), \tilde{x}_{p+1}(k), \dots, \tilde{x}_{n_N}(k))$ при условии $\tilde{x}_p \in \tilde{X}_p$.

Для ХТС без обратных потоков такая процедура поиска нулевого минимума функции F_x полностью соответствует методу покомпонентного спуска в пространстве варьируемых переменных [5], так как в этом случае $\tilde{X}_i \cap \tilde{X}_j = \emptyset$, $i, j = \overline{1, n_N}$. В схемах с обратными потоками для элементов, инцидентных этим потокам, $\tilde{X}_i \cap \tilde{X}_j = \emptyset$, $[i, j] \in I^{(v)}$, где $I^{(v)}$ — множество пар индексов элементов графа, связанных обратными потоками. В этом случае при реализации рассмотренного алгоритма переменные обратного потока, полученные при расчете i -го элемента, целесообразно использовать в качестве начальных значений при расчете j -го элемента схемы.

В пользу выбора такой процедуры расчета элементов ХТС, инцидентных обратным потокам, можно привести следующие доводы. Суммарная функция невязки $F^{(ij)}$ i -го и j -го $[i, j] \in I^{(v)}$, связанных обратным потоком элементов ХТС (с учетом необходимости их согласования), вычисляется так: $F^{(ij)} = F^{(i)} + F^{(j)} + F_n^{(ij)}$, $[i, j] \in I^{(v)}$, где $F^{(i)}$, $F^{(j)}$ — функции невязки соответственно i -го и j -го элементов за счет разбаланса уравнений МБ; $F_n^{(ij)}$ — невязка за счет несогласованности связывающих переменных i -го и j -го элементов.

Величина $F_n^{(ij)}$ определяется соотношением

$$F_n^{(ij)} = \frac{1}{2} \sum_{k=1}^{p_{ij}} (x_k^{(i)} - x_k^{(j)})^2, \quad (5)$$

где p_{ij} — размерность вектора переменных обратного потока, связывающего i -й и j -й элементы; $x_k^{(i)} \in \tilde{X}_i$, $x_k^{(j)} \in \tilde{X}_j$ — переменные обратного потока, варьируемые независимо при расчете i -го и j -го элементов ХТС.

Представим векторы переменных i -го и j -го узлов таким образом: $\tilde{x}^{(i)} = [\tilde{x}_1^{(i)}, \tilde{x}_2^{(i)}]$, $\tilde{x}^{(j)} = [\tilde{x}_1^{(j)}, \tilde{x}_2^{(j)}]$, где $\tilde{x}_1^{(i)} \in (\tilde{X}_i \setminus \tilde{X}_j)$, $\tilde{x}_2^{(i)} \in (\tilde{X}_i \cap \tilde{X}_j)$, $\tilde{x}_1^{(j)} \in (\tilde{X}_j \setminus \tilde{X}_i)$, и вычислим частные производные

от функции $F^{(i)}$ по координатам вектора $\bar{x}^{(i)} \in \bar{X}_i$. Имеем

$$\frac{\partial F^{(i)}}{\partial x_k^{(i)}} = \frac{\partial F^{(i)}}{\partial x_k^{(i)}}, \quad k = 1, \dim \bar{x}^{(i)};$$

$$\frac{\partial F^{(i)}}{\partial x_2^{(i)}} = \frac{\partial F^{(i)}}{\partial x_2^{(i)}} + (x_{2k}^{(i)} - x_{2k}^{(i)}), \quad k = \overline{1, p_i}, \quad x_{2k}^{(i)} \in (\bar{X}_i \cap \bar{X}_j).$$

Погрешность определения вектора производных $\frac{\partial F^{(i)}}{\partial x_2^{(i)}}$ на основании информации о векторе производных $\frac{\partial F^{(i)}}{\partial x_2^{(i)}}$ можно оценить с помощью выражения $\delta = \left\| \frac{\partial F^{(i)}}{\partial x_2^{(i)}} - \frac{\partial F^{(i)}}{\partial x_2^{(i)}} \right\| = \left\| \bar{x}_2^{(i)} - \bar{x}_2^{(i)} \right\|$, где $\bar{x}_2^{(i)}, \bar{x}_2^{(i)} \in \bar{X}_i \cap \bar{X}_j$, а $\| \cdot \|$ — некоторая норма соответствующих векторов.

В соответствии с принятой схемой последовательной минимизации в начальной точке $\bar{x}_0^{(i)}$ при расчете i -го узла выполняются соотношения $x_{2k}^{(i)} - x_{2k}^{(i)} = 0$, $k = \overline{1, p_i}$, поэтому в этой точке $\delta = 0$.

В процессе расчета i -го элемента используется информация только о производных $\frac{\partial F^{(i)}}{\partial x_k^{(i)}}$, $k = \overline{1, \dim \bar{x}^{(i)}}$, поэтому на начальной стадии расчета i -го элемента функция невязки F^i будет уменьшаться. В дальнейшем за счет увеличения рассогласованности связывающих переменных i -й и j -й подсистем величина δ будет расти и движение вдоль градиента, определяемого частными производными $\frac{\partial F^{(i)}}{\partial x_k^{(i)}}$, $k = \overline{1, \dim \bar{x}^{(i)}}$, может привести уже к росту функции невязки $F^{(i)}$, а следовательно, и к росту функции невязки F всей ХТС. Поэтому, если ограничить поиск минимума функции $F^{(i)}$ только несколькими шагами в соответствии с используемой процедурой оптимизации i -го элемента, то можно ожидать, что такой подход в конечном счете позволит нам получить нулевой минимум функции невязки всей системы.

Для выпуклой вниз дифференцируемой функции $F(\bar{x})$ и замкнутых выпуклых множеств $\bar{X}_i$, $i = \overline{1, n_N}$ имеет место соотношение [5] $\lim_{k \rightarrow \infty} \min F(\bar{x}_1(k), \dots, \bar{x}_{n_N}(k)) = \min F(\bar{x}_1, \dots, \bar{x}_{n_N})$, $n_N \leq k$.

В нашем случае выполняются все условия сходимости итерационного процесса, кроме выпуклости функции $F(\bar{x})$. Выпуклая функция, как известно, должна обладать следующим свойством: для любых $\bar{x}^{(1)}, \bar{x}^{(2)} \in R_n$, $s = \dim \bar{x}$, $F(\theta \bar{x}^{(1)} + (1 - \theta) \bar{x}^{(2)}) \leq \theta F(\bar{x}^{(1)}) + (1 - \theta) F(\bar{x}^{(2)})$ (6). Возьмем в качестве векторов $\bar{x}^{(1)}$ и $\bar{x}^{(2)}$ такие два вектора, у которых отличны от нуля только первые две ко-

ординаты x_1 и x_2 , входящие в виде произведения в билинейное уравнение вида (2). Тогда функция невязки F будет иметь вид $F = x_1^2 + x_2^2$. Нетрудно проверить, что для векторов $\bar{x}^{(1)} = (1, 2, 0, \dots, 0)$, $\bar{x}^{(2)} = (2, 1, 0, \dots, 0)$ и $\theta = 0,5$ левая часть выражения (6) равна 7,3125, а правая — 6,5, т.е. условие выпуклости для функции F не выполняется.

Требование выпуклости оптимизируемой функции, гарантирующее ее одноэкстремальность, является значительно более сильным условием, чем условие одноэкстремальности функции, поскольку одноэкстремальность не требует ни непрерывности, ни единственности производной функции в каждой точке. Покажем, что функция невязки уравнений МБ l -го элемента потокового графа не имеет локальных минимумов, отличных от нуля. Будем считать вначале, что рассматриваемый элемент состоит из одного узла. Используя соотношения (1) — (3), вычислим частные производные

$$\frac{\partial F^{(l)}}{\partial x_i^{(l)}}, \quad i = 1 \text{ dim } x^{(l)}$$

$$\frac{\partial F^{(l)}}{\partial m_i} = \begin{cases} f_i^{(l)} + \sum_{j=1}^k f_{(i+1)j}^{(l)} C_{ji}^{(l)}, & i \in I_l^+, \\ -\left(f_i^{(l)} + \sum_{j=1}^k f_{(i+1)j}^{(l)} C_{ji}^{(l)}\right) & i \in I_l^-, \end{cases}$$

$$\frac{\partial F^{(l)}}{\partial C_{ij}} = \begin{cases} f_{(i+1)j}^{(l)} m_j, & i = \overline{1, k}, j \in I_l^+, \\ -f_{(i+1)j}^{(l)} m_j, & i = \overline{1, k}, j \in I_l^-. \end{cases}$$

Стационарные точки функции $F^{(l)}$ являются решениями системы уравнений:

$$f_i^{(l)} + \sum_{j=1}^k f_{(i+1)j}^{(l)} C_{ji}^{(l)} = 0, \quad i \in I_l^+ \cup I_l^-; \quad (7)$$

$$f_{(i+1)j}^{(l)} m_j = 0, \quad i = \overline{1, k}, j \in I_l^+ \cup I_l^-. \quad (8)$$

Поскольку нас интересует ненулевое решение системы балансовых уравнений (1), (2), то $m_j \neq 0$, $j \in I_l^+ \cup I_l^-$, и из выражений (8) следует, что $f_{(i+1)j}^{(l)} = 0$, $i = \overline{1, k}$. Тогда сумма по j в выражении (7) обращается в нуль и, следовательно, $f_i^{(l)} = 0$. Таким образом, стационарные точки функции $F^{(l)}$ соответствуют решениям системы балансовых уравнений, в которых значения функции невязки $F^{(l)}$ равны нулю.

Пусть теперь рассматриваемый элемент состоит из l -го и p -го узлов потокового графа. Выражение функции невязки $F^{(lp)}$ будет иметь вид $F^{(lp)} = F^{(l)} + F^{(p)} + F_n^{(lp)}$, где

$$F^{(l)} = \frac{1}{2} \sum_{i=1}^M (f_i^{(l)})^2, \quad F^{(p)} = \frac{1}{2} \sum_{i=1}^{k_p} (f_i^{(p)})^2,$$

а выражение для невязки $F_n^{(i,p)}$ имеет такую же структуру и смысл, что в (5). Если процедура расчета элементов ХТС построена таким образом, что смежные переменные для каждого узла элемента имеют одно и то же значение на каждой итерации, то, очевидно, $F_n^{(i,p)} = 0$. Если же при расчете элемента смежные переменные, принадлежащие различным узлам, варьируются независимо друг от друга, то в функцию невязки войдет добавка $F_n^{(i,p)}$, характеризующая степень несогласованности смежных переменных.

Покажем, что и в этом случае функция невязки не имеет локальных минимумов, отличных от нулевого. Введем следующие множества: $I^0 = (I_1^+ \cup I_1^-) \cup (I_k^+ \cup I_k^-)$, $I^e = (I_1^+ \cup I_1^-) \cap (I_k^+ \cup I_k^-)$, $I^i = I^0 \setminus I^e$. Вычисляя частные производные функции $F^{(i,p)}$ и приравняв их нулю, получим следующую систему для определения стационарных точек функции невязки:

$$\begin{aligned} \frac{\partial F^{(i,p)}}{\partial m_s^{(i)}} &= -\left(f_{i+1}^{(i)} + \sum_{j=1}^k f_{i+1+j}^{(i)} C_{js}^{(i)}\right) + (m_s^{(i)} - m_s^{(p)}) = 0, \quad s \in I^e; \\ \frac{\partial F^{(i,p)}}{\partial C_{ij}^{(i)}} &= -f_{i+1}^{(i)} m_j + \{C_{ij}^{(i)} - C_{ij}^{(p)}\} = 0, \quad i = \overline{1, k}, \quad j \in I^e; \\ \frac{\partial F^{(i,p)}}{\partial m_s^{(p)}} &= \left(f_i^{(p)} + \sum_{j=1}^k f_{i+1+j}^{(p)} C_{js}^{(p)}\right) - (m_s^{(i)} - m_s^{(p)}) = 0, \quad s \in I^e; \\ \frac{\partial F^{(i,p)}}{\partial C_{ij}^{(p)}} &= f_{i+1}^{(p)} m_j - \{C_{ij}^{(i)} - C_{ij}^{(p)}\} = 0, \quad i = \overline{1, k}, \quad j \in I^e. \end{aligned} \quad (9)$$

Если индексы $s, j \in I^b$, то производные определяются выражениями, аналогичными (7), (8), с соответствующими верхними индексами. Для простоты записи выражения (9) записаны в предположении, что все смежные потоки являются выходящими из i -го узла графа.

Анализируя соотношение (7) — (9), можно сделать вывод о том, что решениями данной системы являются только те значения переменных, при которых выполняются уравнения балансов и условия согласования смежных переменных. Действительно, из выражений $f_{i+1}^{(i)} m_i = 0$, $f_{i+1}^{(p)} m_i = 0$, $i = \overline{1, k}$, $j \in I^b$, следует, что $f_{i+1}^{(i)} = 0$, $f_{i+1}^{(p)} = 0$, $i = \overline{1, k}$, $j \in I^b$ (10). Подставляя (10) в выражения (7), получаем $f_i^{(i)} = 0$, $f_i^{(p)} = 0$. Тогда из уравнений (9) следует, что в стационарной точке должны выполняться соотношения $m_s^{(i)} = m_s^{(p)}$, $C_{ij}^{(i)} = C_{ij}^{(p)}$, $i = \overline{1, k}$, $s, j \in I^e$. Аналогичным образом можно показать, что функция невязки элемента из любого числа узлов потокового графа не имеет локальных экстремумов, отличных от нуля. Этот вывод позволяет для расчета элементов ХТС использовать сравнительно хорошо разработанные алгоритмы локального поиска, так как в данном случае любой локальный минимум функции невязки является одновременно глобальным.

Покажем, что условие равенства нулю функции невязки на 0-узле является только необходимым условием равенства нулю невязок на всех элементах ХТС при условии согласованности переменных смежных элементов ХТС. Действительно, непосредственной проверкой можно убедиться, что при условии согласованности всех переменных системы координаты вектора невязки 0-узла могут быть представлены следующим образом:

$$f_i^{(0)} = - \sum_{j=1}^N f_j^{(j)} = 0, \quad i = \overline{1, k_j} \quad (11)$$

где N — число узлов потокового графа, а k_j — число различных типов уравнений балансов ХТС. Из равенства

$$F = \frac{1}{2} \sum_{i=1}^{k_j} F^{(i)} = \frac{1}{2} \sum_{j=1}^N \sum_{i=1}^{k_j} (f_i^{(j)})^2 = 0$$

следует, что $f_i^{(j)} = 0$, $j = \overline{1, N}$, $i = \overline{1, k_j}$ (12). Тогда на основании соотношения (11) имеем $f_i^{(0)} = 0$, $i = \overline{1, k_j}$, $j = \overline{1, N}$ (13). Из равенств (12), однако, не следуют равенства (13), поэтому соотношения (13), при которых $F^{(0)} = 0$, являются только необходимыми, но не достаточными условиями равенства нулю функции невязки F всей ХТС.

Несо согласованность смежных переменных схемы эквивалентна, очевидно, появлению в системе дополнительных вершин-источников и вершин-стоков, что приводит в общем случае к еще большему разбалансу уравнений на 0-узле.

Целесообразность включения в вычислительную схему последовательно-декомпозиционного метода задачи минимизации функции невязки F^0 0-узла можно обосновывать следующими соображениями. Как правило, на 0-узле замыкаются значительное количество вершин-источников и вершин-стоков потокового графа ХТС. И хотя условие равенства нулю функции F^0 является только необходимым для выполнения равенства $F=0$, тем не менее, минимизируя невязку на 0-узле, можно ожидать, что полученные в результате расчета 0-узла, значения большого числа переменных будут являться хорошими начальными приближениями при расчете остальных элементов ХТС.

Разработанный последовательно-декомпозиционный метод положен в основу создания универсальной машинной программы расчета балансовых уравнений технологических комплексов с обратными потоками, которая в настоящее время внедряется в практику технологического проектирования сложных химических производств.

Список литературы: 1. Кафаров В. В., Перов В. Л., Мешалкин В. П. Принципы математического моделирования химико-технологических систем. — М.: Химия, 1974. — 344 с. 2. Островский Г. М., Воронин Ю. Н. Моделирование сложных химико-технологических схем. — М.: Химия, 1975. — 312 с. 3. Зозула А. Ф., Асеев Г. Г. Циклический метод анализа и расчета потоковых схем в системах

автоматизированного проектирования.—АСУ и приборы автоматики, 1979, вып. 30, с. 21—28. 4. Зозуля А. Ф., Асеев Г. Г. Об одном методе моделирования сложных потоковых схем.—АСУ и приборы автоматики, 1979, вып. 52, с. 60—64. 5. Математические методы исследования операций/ Ю. М. Ермаков, Н. Н. Лешко, В. С. Михалевич, В. Н. Тюптя.—К.: Высш. школа, 1979.—312 с.

Поступила в редакцию 26.01.82.

УДК 681.5.015

А. Д. ТЕВЯШЕВ, В. Д. ГРЕБЕННИК

О ВЫБОРЕ РАЦИОНАЛЬНОЙ СТРУКТУРЫ МОДЕЛИ УСТАНОВИВШЕГОСЯ ПОТОКОРАСПРЕДЕЛЕНИЯ В ИНЖЕНЕРНОЙ СЕТИ

Введение. При решении задач оперативного планирования режимов транспорта и распределений целевого продукта в инженерных сетях в качестве модели объекта управления используется модель установившегося потокораспределения инженерной сети [1].

При этом инженерная сеть представляется в виде графа $G(V_0, E_0)$, отражающего структуру объекта управления, где $V_0 = \{0, 1, 2, \dots, v_0\}$ — множество вершин графа сети; $E_0 = \{1, 2, \dots, e_0\}$ — множество дуг графа. Модель установившегося потокораспределения, соответствующая графу $G(V_0, E_0)$, имеет вид [2]:

$$\sum_{i \in M_i} b_{1,i} \psi_i(P_{1,i}, P_{i,n}, q_i, \beta_i) + \psi_r(P_{r,n}, P_{r,n}, q_r, \beta_r) = f_r(q_i, \beta_i) = 0, \quad i \in E, \quad r \in M_2; \quad (1)$$

$$\sum_{i \in M} b_{1,i} \psi_i(P_{1,i}, P_{i,n}, q_i, \beta_i) = f_r(q_i, \beta_i), \quad i \in E, \quad r \in N_2; \quad (2)$$

$$q_i = \sum_{i \in E_1} b_{1,i} q_r, \quad i \in E_1, \quad (3)$$

где ψ — определенные в D нелинейные, монотонно возрастающие или монотонно убывающие непрерывные, вместе со своими частными производными, функции; M — множество индексов реальных дуг графа сети; V_0 — множество индексов узлов сети; $P_{1,i}$ — давление в начале i -й дуги; $P_{i,n}$ — давление в конце i -й дуги; β_i — вектор параметров i -го элемента; q_i — расход целевого продукта по i -й дуге; E_0 — множество индексов всех дуг графа сети; E_1, E_2 — множество индексов ветвей и хорд графа сети $E = E_1 \cup E_2$; $E_1 = M_1 \cup N_1$; $E_2 = M_2 \cup N_2$; M_1 — множество индексов реальных ветвей дерева; M_2 — множество индексов реальных хорд дерева; N_1 — множество индексов фиктивных ветвей графа сети; N_2 — множество индексов фиктивных хорд дерева; $b_{1,i}$ — элемент цикломатической матрицы B_1 для ветвей дерева графа сети;

$$P_r = \begin{cases} P_{r_n}, & \text{если фиктивная хорда соединяет выход сети} \\ & \text{с фиктивной вершиной;} \\ P_{r_k}, & \text{если фиктивная хорда соединяет вход сети} \\ & \text{с фиктивной вершиной;} \end{cases}$$

$$P_l = \begin{cases} P_{l_n}, & \text{если фиктивная ветвь дерева соединяет выход} \\ & \text{сети с нулевой точкой;} \\ P_{l_k}, & \text{если фиктивная ветвь дерева соединяет вход} \\ & \text{сети с нулевой точкой.} \end{cases}$$

Цикломатическая матрица B_1 определяется только графом сети $G(V_0, E_0)$ и не зависит от векторов параметра β_i .

Решение задач оперативного планирования режимов функционирования инженерных сетей основано на многократном проведении гидравлических расчетов инженерных сетей, т. е. решении системы нелинейных алгебраических уравнений (1)–(3) при соответствующем задании граничных условий. При этом время решения системы уравнений (1)–(3) и требуемый объем памяти ЭВМ определяется мощностью множеств V_0 и E_0 .

Для систем водо-, газо- и теплоснабжения крупных городов количество узлов и дуг соответствующих сетей достигает десятков и сотен тысяч, а допустимое время решения задач оперативного планирования режимов не должно превышать одного часа [2].

В таких условиях целесообразно разработать методы упрощения структуры модели объекта, позволяющие получить модели, отвечающие требованиям как по точности результатов, так и по времени решения задач управления.

В зависимости от уровня априорной информации о структуре и параметрах объекта управления, возможностей получения необходимых экспериментальных данных, требований к характеристикам модели возможны два основных подхода к постановке и решению задач структурной идентификации — детерминированный и вероятностный.

Детерминированный подход.

Задача 1. Пусть в результате обследования объекта управления удалось установить: структуру объекта в виде графа $G(V_0, E_0)$; параметры всех реальных участков β_{0i} , $i \in M$, среднее значение расходов в каждом j -м, $j \in V_0$ узла инженерной сети $\bar{q}_{0j}$. Для графа $G(V_0, E_0)$ имеется процедура сюръективного отображения множества узловых расходов $\bar{q}_0 = \{\bar{q}_{01}, \bar{q}_{02}, \dots, \bar{q}_{0n}\}$ на множество давлений в узлах $P_0 = \{P_{01}, P_{02}, \dots, P_{0n}\}$, т. е. $\bar{q}_0 \xrightarrow{f} P_0$, где f — алгоритмически заданная функция (задача гидравлического расчета инженерной сети).

Кроме того, задано подмножество $V' \subseteq V_0$ вершин графа, которые должны войти в модель, а также общее количество вершин графа модели $v = \text{card } V$; $V' \subseteq V \subseteq V_0$.

Необходимо найти такую структуру $G(V, E)$ и параметры β модели, при которых максимальная степень расхождения между значениями давления $\tilde{P}_i, i \in V_0$ в узлах сети, найденными по полной модели $G(V_0, E_0)$, и давлениями P_i в тех же узлах, найденными по упрощенной модели $G(V, E)$, не превосходит некоторого наперед заданного значения $\epsilon: \max_i \|\tilde{P}_i - P_i\| \leq \epsilon, i \in V$ (4), при

условии, что $\tilde{P}_k = P_k, k \in V_1 \subset V_0, \tilde{q}_j = q_j$, где V_1 — множество индексов вершин графа сети, являющихся входами и выходами активных элементов; $j \in L$ — множество индексов дуг с активными элементами.

Задача 2. В условиях задачи 1 вместо вредных значений расходов $\tilde{q}_j$ в каждом j -м, $j \in V$, узле инженерной сети известен весь диапазон изменения расхода от $q_j^{\min}$ до $q_j^{\max}$.

В этом случае в качестве критерия адекватности используется условие (4), которое должно выполняться для произвольной комбинации узловых расходов на активных элементах $q_{i_0}^{\min} < q_i < q_{i_0}^{\max}, i \in L: \max_i \max_{\vec{q}} \|\tilde{P}_i - P_i\| \leq \epsilon, i \in V, \vec{q} \in Q$ (5), где Q — допустимая

область изменения вектора расходов $\vec{q}$. В общем случае решением задач (1) и (2) является некоторая область Ω значений $E, \vec{q}$ и β . Единственное решение может быть получено путем введения дополнительного критерия. В качестве такого критерия можно выбрать время решения задачи гидравлического расчета.

Задача 3. В условиях задачи (1) или (2) необходимо найти такие значения $E, \vec{q}, \beta$, которые удовлетворяют ограничениям (4) или (5) и доставляют минимум функции $\tau(G(V, E)): \tau_0(G(V, E)) = \min_{V, E, \vec{q}, \beta \in \Omega} \tau(G(V, E))$ (6).

Выбор единственного решения из множества Ω может быть произведен с целью выбора структуры, минимизирующей максимальное значение нормы отклонения (4).

Задача 4. В условиях задач 1, 2 необходимо найти такие $V, E, \vec{q}, \beta$, которые обеспечивают $\min_{V, E, \vec{q}, \beta \in \Omega} \max_i \|\tilde{P}_i - P_i\|$ (7).

Стохастический подход. В рассмотренных выше постановках предполагалось, что известны истинные значения расходов в узлах и сопротивлений в дугах, что в реальных ситуациях бывает крайне редко. При использовании моделей для решения задач планирования и оперативного управления потокораспределением расходы в узлах и сопротивления реальных участков сети представляют собой случайные процессы. В связи с этим вектор давлений в узлах как функция случайных аргументов является случайным вектором. Истинные значения давлений в узлах сети

как реализации случайных процессов не будут совпадать с расчетными, полученными ранее на модели, структура которой задана полным графом $G(V_0, E_0)$.

В такой ситуации требования к модели должны быть сформулированы с учетом вероятностного характера вектора давлений в узлах сети из множества V_0 . В качестве таких требований может быть задана вероятность того, что истинное значение давления в i -м узле $P_i(t)$ в момент времени t окажется меньше расчетного $\tilde{P}_i(t)$. При заданных структуре $G(V, E)$ законах распределения $f(\vec{q})$ и $f(\vec{\beta})$ можно найти указанную вероятность для i -го узла $P\{(P_i(t) - \tilde{P}_i(t)) \leq 0\}$.

Представление структуры модели в виде упрощенного графа $G(V, E)$ внесет дополнительную неопределенность, связанную с использованием неадекватного графа реальной структуры объекта.

Важно подчеркнуть, что характер неопределенности, вносимой случайной природой исходных данных ($f(\vec{q})$, $f(\vec{\beta})$) и упрощенной структурой модели, одинаков. Он выражается в изменении закона распределения случайной величины $\Delta(t) = \tilde{P}_i(t) - P_i(t)$. Поэтому задача выбора рациональной структуры модели $G(V, E)$ формулируется следующим образом.

Задача 5. Известны $G(V_0, E_0)$, V , $f(\vec{q}) = f(q_1, q_2, \dots, q_n)$, $f(\vec{\beta}) = f(\beta_1, \beta_2, \dots, \beta_n)$. Требуется найти $G(V, E)$, E , $\vec{q} = \{q_1, q_2, \dots, q_n\}$, $\vec{\beta} = \{\beta_1, \beta_2, \dots, \beta_n\}$ такие, что $\max_i P\{(P_i(t) - \tilde{P}_i(t)) \leq 0\} \leq \alpha$ (8). Законы распределения $f(\vec{q})$ и $f(\vec{\beta})$ могут быть такими, что ограничения (8) не выполняются даже при использовании в качестве структуры модели полного графа $G(V, E)$. В таком случае необходимо уменьшить дисперсии оценок компонент вектора $\vec{q}$ (уменьшение дисперсии измерений, прогноза и т. п.) и вектора $\vec{\beta}$ (повышение требований к точности системы идентификации).

Если же требование (8) может быть удовлетворено, то представляется естественным сформулировать следующую задачу.

Задача 6. В условиях задачи (5) необходимо найти такую структуру модели $G(V, E)$, E , $\vec{q}$, $\vec{\beta}$, которая обеспечит выполнение требований (8) и доставит минимум времени решения задач (1-3): $\tau_0(G(V, E)) = \min_{E, \vec{q}, \vec{\beta}} \tau(G(V, E))$. В задачах оперативного

управления, где время решения является особенно ощутимой платой за точность решения, возникает задача поиска оптимальной структуры модели $G(V, E)$.

Действительно, если точность исходных данных $\vec{q}$ и $\vec{\beta}$ опре-

деляется временем прогноза соответствующих компонент векторов $\vec{q}$ и $\vec{\beta}$, а время прогноза $\tau(G(V, E))$ — временем решения задач (1—3), то налицо две противоположных тенденции.

С одной стороны, использование более подробной модели приводит к повышению точности результата, с другой, — к увеличению времени прогноза компонент векторов $\vec{q}$ и $\vec{\beta}$ и, как следствие, уменьшению точности представления исходных данных.

В такой ситуации задача поиска оптимальной структуры формируется так:

Задача 7. Дано: $G(V_0, E_0)$, $f(\vec{q}, \tau)$, $f(\vec{\beta}, \tau)$. Найти $G(V, E)$, $V, E, \vec{q}, \vec{\beta}$, которые обеспечивают $\min_{V, E, \vec{q}, \vec{\beta}} \max_i P\{(\tilde{P}_i(t) - P_i(t))\}$, где

$f(\vec{q}, \tau)$ и $f(\vec{\beta}, \tau)$ — плотности распределения векторов $\vec{q}, \vec{\beta}$ в зависимости от времени прогноза τ .

Сформулированные задачи относятся к классу задач комбинаторного типа на ориентированных графах, для решения которых могут быть использованы методы направленного перебора [3]. Процесс решения этих задач предполагает решения ряда таких подзадач: определение количества узлов и дуг модели объекта управления; определение степени связности графа $G(V, E)$; оценка параметров и состояний (узловых расходов) модели; проверка адекватности модели.

Основные трудности при решении этих задач связаны с построением процедуры целенаправленной эволюции графа модели сети. При этом определяющую роль играет способ выбора количества и местоположения вершин графа сети. Для получения оценок компонент вектора $\vec{q}$ и $\vec{\beta}$ используем стандартные приемы эквивалентирования и статистического анализа экспериментальных данных [2, 4, 5].

Получение зависимостей $\tau(G(V, E))$ не представляет значительных трудностей при фиксированном графе сети $G(V, E)$. Такие зависимости получены для различных алгоритмов решения задачи гидравлического расчета водопроводной сети [1].

При решении задач 5, 6, 7 необходимо преодолеть дополнительные к задачам 1—4 трудности, связанные с определением законов распределения $f(\vec{P}) = f(P_1, P_2, \dots, P_n)$ при известных $f(\vec{q})$ и $f(\vec{\beta})$.

Список литературы: 1. Ефодкилов А. Г., Тевяшев А. Д. Оперативное управление потокораспределением в инженерных сетях. — Х.: Высшая школа, 1980. — 143 с. 2. Ефодкилов А. Г., Дубровский В. В., Тевяшев А. Д. Потокораспределение в инженерных сетях. — М.: Стройиздат, 1979. — 199 с. 3. Фриш Г., Фриш Н. Сети, связь и потоки. — М.: Связь, 1978. — 448 с. 4. Химмельблау Д. Анализ процессов статистическими методами. — М.: Мир, 1973. — 547 с. 5. Сухирев М. Г., Ставроцкий Е. Р. Расчет систем транспорта газа с помощью вычислительных машин. — М.: Недра, 1971. — 206 с.

Поступила в редакцию 19.02.82.

А. Н. ЛИТВИНОВ

МАТРИЧНОЕ ПРЕДСТАВЛЕНИЕ ЗАДАЧИ ОПТИМАЛЬНОГО РАЗМЕЩЕНИЯ ОТКЛЮЧАЮЩИХ УСТРОЙСТВ НА ИНЖЕНЕРНЫХ СЕТЕВЫХ СИСТЕМАХ

Данная задача является составной частью общей задачи оптимального оперативного управления потокораспределением в инженерных сетевых системах при аварийных режимах. Отказ (авария) линейных участков инженерной сети приводит к значительным ущербам вследствие прямых потерь целевого продукта, изменений режимов функционирования сети, возникновения дефицита целевого продукта в районе аварии и районах, прилегающих к аварийному участку, а в некоторых случаях — и во всей системе. Ущерб может быть нанесен самим целевым продуктом — затопления водой, взрывы газа и т. д.

Оптимальное оперативное управление потокораспределением в таких ситуациях сводится к поиску рациональной структуры сети путем коммутации отключающих устройств с целью исключения из нее аварийного участка и минимизации некоторого функционала, характеризующего ущерб, возникающие в системе из-за аварий. Такое управление возможно лишь в случае оптимального размещения отключающих устройств, а вследствие того, что эти устройства дорогостоящи, то количество их сводится к обоснованному минимуму.

Современные инженерные сетевые системы содержат большое количество взаимосвязанных участков. Сети в процессе эксплуатации обустраиваются внутренними переключками и отводами, в результате чего приобретают сложные конфигурации с большим количеством рассредоточенных мест отбора целевого продукта. Поэтому решение задач оптимального размещения отключающих устройств немисливо без применения средств современной вычислительной техники, что в свою очередь предъявляет специальные требования к математическим моделям такого размещения.

Наиболее удобной формой математического представления графа инженерной сети и его элементов на этапе формулировки проблемы и отыскания алгоритма ее решения является матричная форма [1], которой мы и будем пользоваться. При необходимости можно легко перейти к списочному или теоретико-множественному представлению [1].

Рассмотрим некоторый конечный ориентированный связный линейный граф, отображающий топологию какой-либо инженерной сетевой системы, с e ребрами (ветвями) и v вершинами (узлами). Математически линейный граф однозначно отображается матрицей инцидентности [2], $A_s = [a_{ij}]$; $i = \overline{1, v}$; $j = \overline{1, e}$.

К каждой ветви графа инженерной сети присоединены потребители, несущие определенные материальные ущербы в случае прекращения подачи целевого продукта в данную ветвь. Поскольку речь идет именно об оптимальном размещении отключающих устройств, то к ветвям графа сети необходимо подходить дифференцированно, т. е. присвоить каждой ветви графа приоритетное число m_j , $j = \overline{1, e}$; $m_j \in N$, которое будет характеризовать величину ущерба от недопоячки целевого продукта потребителям, питающимся от j -й ветви. Такой дифференцированный подход необходим для того, чтобы было возможно оценить степень ответственности каждой ветви графа сети по отношению к другим.

В соответствии с этим построим матрицу приоритетных весов $W = \{w_{ij}\}$; $i = \overline{1, v}$; $j = \overline{1, e}$, каждый элемент w_{ij} которой определим следующим образом: $w_{ij} = a_{ij} m_j$.

Если полученную матрицу W , характеризующую в совокупности степень ответственности каждой ветви, присоединенной к данному узлу (i -я строка матрицы соответствует i -му узлу графа), умножим на единичный вектор-столбец $\bar{h}$ размерности e , то получим v -мерный вектор-строку $\bar{q} = W\bar{h}$, характеризующий степень ответственности каждого узла графа сети (i -я координата вектора $\bar{q}$ соответствует i -му узлу) $\bar{q} = W\bar{h}$.

В связи с тем что приведенная нами задача включает в себя не только нахождение оптимальных координат размещения отключающих устройств, но и определение их минимального количества, сформулируем следующую лемму, определяющую верхнюю границу количества отключающих устройств, устанавливаемых на графе инженерной сети. Лемма доказана, однако в связи с ограниченным объемом статьи доказательство не приводится.

Лемма 1. Для локализации любой ветви графа инженерной сети и соблюдения условия питания целевым продуктом всех остальных ветвей сети отключающие устройства необходимо размещать следующим образом.

1. В питающем узле отключающее устройство должно быть в каждой ветви, инцидентной этому узлу.

2. Во всех остальных $(v - 1)$ узлах графа сети, каждому из которых инцидентно по n_i ветвей, где $i = \overline{1, (v - 1)}$; $n_i \in N$ ставится по $(n_i - 1)$ отключающих устройств (в каждой ветви, инцидентной i -му узлу, не более одного) в произвольном порядке.

Следствие. Если граф инженерной сети содержит v узлов и e ветвей ($e > v$), то верхним пределом количества отключающих устройств будет число R , равное $R = 2e - (v - 1) = 2e - v + 1$.

Замечание. Лемма 1 справедлива лишь в том случае, если матрица главных сечений Q графа сети не содержит ни одной строки, в которой бы количество ненулевых элементов было меньше двух. Замечание очевидно, поэтому приводится без до-

казательств. Анализ реально существующих инженерных сетей показал, что для любой из них в Q_* не содержится ни одной такой строки.

Значение верхнего предела количества отключающих устройств будет использовано для оценки эффективности разработанных алгоритмов оптимального размещения отключающих устройств на инженерных сетевых системах.

Конечной целью поставленной задачи является определение координат размещения отключающих устройств по некоторому, нами заданному критерию оптимальности, в котором основную роль должны играть приоритетные веса ветвей графа сети. С этой целью введем такие понятия, как пространство состояний (размещений) отключающих устройств S и обобщенная матрица возможных ущербов графа сети M .

Если i -му узлу графа сети инцидентно n_i ветвей $i = \overline{1, v}$; $n_i \in N$, то в таком узле, очевидно, возможно $K_i = 2^{n_i}$ комбинаций размещения отключающих устройств (начиная с варианта, при котором в каждой ветви, инцидентной i -му узлу, стоит отключающее устройство и, кончая вариантом, когда ни в одной из ветвей данного узла отключающего устройства нет). Для каждого i -го узла графа сети построим матрицу $S_i = [s_{kr}^i]$, где $k = \overline{1, 2^{n_i}}$; $r = \overline{1, e}$; $i = \overline{1, v}$, которую назовем матрицей возможных состояний (размещений) отключающих устройств в i -м узле. Элементы каждой матрицы S_i определяются из условий:

$$s_{kr}^i = \begin{cases} 1, & \text{если в } r\text{-й ветви, инцидентной } i\text{-му узлу, в } k\text{-й} \\ & \text{комбинации размещения отключающих устройств} \\ & \text{для } i\text{-го узла стоит отключающее устройство;} \\ 0, & \text{если не стоит отключающее устройство, или } r\text{-я} \\ & \text{ветвь не инцидентна } i\text{-му узлу.} \end{cases}$$

Каждая из построенных таким образом матриц S_i имеет размерность $2^{n_i} \times e$.

Для единого подхода при составлении комбинаций размещения отключающих устройств в каждом узле графа сети условимся перебирать их, начиная с комбинации, при которой в каждой ветви, инцидентной рассматриваемому узлу, стоит отключающее устройство. В последующих комбинациях количество отключающих устройств будем постепенно уменьшать, начиная с ветвей со старшими номерами, а последнюю комбинацию приведем в соответствие с той, когда ни в одной ветви рассматриваемого узла нет отключающего устройства.

Очевидно, что при таком переборе комбинаций в каждой матрице S_i первая строка $[s_{1r}^i]$ будет в точности идентична i -й строке матрицы инцидентий графа сети A_* .

Множество всех матриц возможных состояний размещения отключающих устройств S_i составляет пространство состояний графа сети: $S = \{S_i; i = \overline{1, v}\}$.

Остается только выбрать вариант, удовлетворяющий критерию оптимальности. Для этой цели необходима обобщенная матрица возможных ущербов M графа сети, которую введем следующим образом.

Если матрицу S_i размерности $2^{n_i} \times e$ умножим на вектор-столбец $[w_{i,1}, w_{i,2}, \dots, w_{i,e}]^T$, составленный из i -й строки матрицы приоритетных весов $W = [w_{i,j}]$, где $i = \overline{1, v}$; $j = \overline{1, e}$, то получим вектор-строку размерности 2^{n_i} , который определим как i -ю строку обобщенной матрицы возможных ущербов M графа инженерной сети.

$$S_i \begin{bmatrix} w_{i,1} \\ w_{i,2} \\ \vdots \\ w_{i,e} \end{bmatrix} = [\mu_{i,1}, \mu_{i,2}, \dots, \mu_{i,2^{n_i}}].$$

Построенная таким образом матрица M имеет размерность $v \times \max [2^{n_i}; i = \overline{1, v}]$. В строках матрицы M , размерность которых $2^{n_i} < \max [2^{n_i}; i = \overline{1, v}]$, недостающие элементы добавим нулями. Каждому элементу i -й строки матрицы M соответствует вполне определенная комбинация размещения отключающих устройств в i -м узле графа инженерной сети. Величина этого элемента характеризует степень предотвращения ущерба в аварийной ситуации.

Если произведена выборка по одному элементу из каждой строки матрицы M , то тем самым уже автоматически получают заданы координаты размещения и количество отключающих устройств (из пространства состояний S) в каждом узле графа инженерной сети.

Сумма значений всех элементов выборки характеризует степень предотвращения ущерба при данной системе размещения отключающих устройств.

Таким образом, задача оптимального размещения отключающих устройств на инженерных сетевых системах свелась к задаче нахождения алгоритмов выборки элементов из каждой строки обобщенной матрицы возможных ущербов M , построенной и обоснованной в данной работе.

Список литературы: 1. Ефодимов А. Г., Дубровский В. В., Тевляшев А. Д. Потокораспределение в инженерных сетях. — М.: Стройиздат, 1979. — 195 с.
2. Берге К. Теория графов и ее применение. — М.: Изд-во иностр. лит., 1962. — 198 с.

Поступила в редакцию 28.12.82.

В. А. ПОПОВ, канд. техн. наук, И. В. ЧУМАЧЕНКО

МАШИННЫЙ АНАЛИЗ ОПЕРАЦИОННОГО И УПРАВЛЯЮЩЕГО УСТРОЙСТВ НА БОЛЬШИХ ИНТЕГРАЛЬНЫХ СХЕМАХ. Ч. I

Постановка задачи. Разработка микропроцессорных вычислительных структур (МВС) состоит из следующих основных этапов:

1. Разработка технического задания (ТЗ) на основе изучения конкретной системы управления объектом, результатом которого является определение основных требований к МВС, список алгоритмов обработки информации, аппаратурный состав системы обработки информации, требования по техническим, эксплуатационным, конструктивным параметрам (разрядность, формат слов, рабочий цикл типовой операции, система команд и др.).

2. Составление функциональной схемы МВС на базе выбранного состава кристаллов и основных блоков (ОЗУ, ПЗУ, ПЛМ, ЗУМК, ЦПЭ, БПП, коммутатора и т. д.) с учетом требований п. 1.

3. Составление микропрограмм системы команд в условиях функциональной схемы п. 2.

4. Отработка микрокоманд с помощью моделирования и макетирования.

5. Разработка общего и специального математического обеспечения с учетом п. 1—4.

6. Стендовые испытания макета.

В данной методике предлагается процедура формализованного проектирования операционного устройства МВС, позволяющая автоматизировать ряд этапов. Основная идея состоит в представлении заданного полного алгоритма обработки информации A ($a \in A$, например a — элемент системы команд) через функционалы кристаллов K ($v \in V$, $K(v)$ — описание кристалла), что позволяет оценить степень соответствия обрабатываемых алгоритмов A на заданном наборе кристаллов V , где степень соответствия оценивается числом кристаллов, их номенклатурой, временем реализации алгоритмов. Количественные оценки соответствия A и V определяются путем машинного перебора элементов алгоритмов (микрокоманд, логандов) и сравнения их с микрокомандами (логандами кристаллов) [4].

Принимаются следующие ограничения. Рассматривается только процессорная часть (обработка микрокоманд), а система памяти и управления учитывается в связи с их общей работой в МВС. Предполагается в наличии банк кристаллов для выбора из них рациональных кристаллов (типа ЦПЭ). Применяемая методика позволяет выявить рациональные варианты.

Основные этапы машинного проектирования логических структур на БИС и МП. Выделяются этапы синтеза дискретных устройств на БИС: разработка технического задания (формирование вида исходных данных и получаемых результатов); составление формализованного ТЗ; анализ и выбор элементной базы; построение логической структуры на БИС и МП.

Решение задачи проектирования предполагает формальное описание всех этапов при помощи математического аппарата РСА, работающего на различных уровнях детализации. Каждый этап проектирования необходимо заканчивать регулярным выражением в строго определенном виде, удобном для работы с ним на последующих этапах.

На этапе разработки и анализа ТЗ необходимо получить регулярное выражение алгоритмов преобразования информации R , которое описывается посредством элементарных операций. Составляется список P , в котором элементарные операции, необходимые для реализации заданных алгоритмов, встречаются не более одного раза. Таким образом, набор P есть список необходимых микрокоманд предполагаемой структуры на БИС. На любое предполагаемое устройство могут накладываться технические ограничения, в данной методике отраженные в виде набора характеристик.

Таким образом, на уровне формализации ТЗ требования заказчика должны быть представлены множествами

$$Q = \{R, P(R), H1\}.$$

Функционирование любого кристалла БИС и МП можно описывать при помощи РСА типа K , компонентами которой будут операторы списка микрокоманд. Для увеличения быстродействия устройства необходимо выполнение последовательного участка алгоритма на одной БИС. Некоторое последовательное количество операций (больше одной) алгоритма, начинающееся с подготовки данных для реализации этого участка в кристалле и заканчивающееся выдачей результата из последнего, будем называть логандом. Выполнение логандов в алгоритмах способствует выполнению быстродействия устройства за счет ликвидации времени на пересылку информации между кристаллами и на обращение к ОЗУ. Таким образом, составление регулярных выражений для логандов L следует из анализа выражения K . Аналогично, как и для уровня анализа технического задания, составляется список технических характеристик одного кристалла $H2$. На основании указанного выше один кристалл БИС или МП можно описать множеством

$$S = \{K, L(K), H2\}.$$

В связи с тем что для действительного выбора элементной базы необходимо наличие большого количества БИС и МП, требуется составление банка кристаллов, представляющего собой

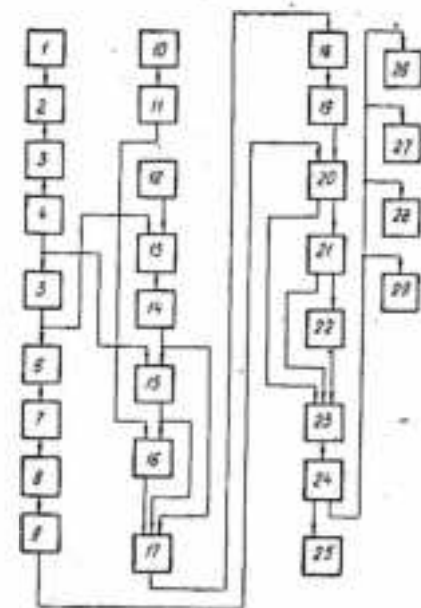
множество $B = \bigcup S_i = \bigcup \{K_i, L_i(K), H2_i\}$, где K_i — количество кристаллов в банке.

Определение набора кристаллов для заданного класса задач производится сравнением регулярных выражений Q и $\bigcup S_i = B$.

По результатам анализа можно получить основные характеристики разработанного устройства: набор кристаллов БИС и МП в устройстве ($v \in B$); частоту использования списка микрокоманд каждого кристалла ($\mu_i(P_i)$; $P_i \in P$); операции, не реализованные

на полученном наборе кристаллов ($P_i \in P$, $P_i \in P'$, $P_i \in P''$, $P' \cup P''$, $P' \cap P'' = \emptyset$); время решения задачи R : время $T = \sum_i t_i K_i(P_i)$ или среднее время $T_{\text{ср}} = \sum_i \mu_i(P_i) t_i$.

Проектирование логического устройства на БИС можно произвести следующим образом. Каждая операция из R выполняется в определенный момент функционирования алгоритма. Производится преобразование РСА R таким образом, что группа операторов на списке микрокоманд набора БИС заменяется на функционалы кристаллов. Полученная РСА $N(R)$ описывает работу логического устройства, но предполагает избыточность, так как каждый кристалл БИС встречается в N столько раз, сколько операций из его списка



Блок-схема: методика анализа устройств на БИС

ка микрокоманд использовалось в R . Из анализа временной диаграммы алгоритмов можно сформулировать требования к количеству кристаллов БИС и МП, реализующих поставленные задачи. Накладыванием полученных требований на РСА $N(R)$ получаем РСА F , которая отражает организацию структуры устройства. Операторы из РСА N , не являющиеся функционалами, могут быть использованы для синтеза дополнительных схем из элементов средней степени интеграции, а также в качестве списка микрокоманд для БИС или МП частного применения.

На рисунке представлена последовательность действий (блоков) методики анализа устройств на БИС: 1 — задание структуры и списка микрокоманд БИС; 2 — формализация структуры

БИС на уровне регистровых передач; 3 — формализация списка микрокоманд каждой БИС; 4 — составление логандов для БИС; 5 — составление сквозного списка микрокоманд для всех БИС; 6 — определение соответствия сквозного списка микрокоманд каждой конкретной БИС; 7 — кодирование сквозного списка микрокоманд операторами регулярных схем алгоритмов; 8 — представление логандов операторами сквозного списка; 9 — составление банка данных элементной базы; 10 — выбор основных технических и информационных характеристик БИС; 11 — кодирование основных характеристик БИС операторами регулярных схем алгоритмов; 12 — задание алгоритмов обмена или обработки информации на проектируемое устройство в произвольной форме; 13 — представление алгоритмов в сквозном списке микрокоманд; 14 — составление выражения алгоритмов задания; 15 — представление алгоритмов в логандах БИС; 16 — составление выражения технических характеристик на разрабатываемую систему; 17 — полное выражение задания на разрабатываемую систему; 18 — минимизация полного выражения технического задания; 19 — преобразование РСА в вид, удобный для работы в ППП; 20 — подбор элементной базы по критерию технических характеристик; 21 — подбор элементной базы по составу логандов; 22 — подбор элементной базы по микрооператорам; 23 — определение количества БИС и микропроцессоров каждого типа; 24 — список нереализуемых функций на БИС; 25 — определение порядка соединений кристаллов в разрабатываемой системе; 26 — синтез устройства управления на основе нереализованных функций; 27 — настройка программируемой логической матрицы; 28 — синтез устройств средней степени интеграции; 29 — разработка заказных БИС частого применения. Этапы разработки анализа и формализации технического задания (бл. 1, 2, 3, 10, 12) должны производиться заказчиком устройства совместно с разработчиком устройства. Подготовка данных для пакета прикладных программ занимает блоки 4—9, 11, 13—16. Остальная часть методики (бл. 17—19) может быть формализована и заменена пакетом прикладных программ (ППМ).

Список литературы: 1. Казан В. М., Стасин В. В. Микропроцессоры в цифровых системах. — М.: Энергия, 1979. — 192 с. 2. Маклаин Д. Микропроцессоры: технология, архитектура и применение/Пер. с англ. — М.: Энергия, 1979. — 224 с. 3. Хилбурн Дж., Джулич П. Микропроцессоры и микро-ЭВМ/Пер. с англ. — М.: Мир, 1979. — 463 с. 4. Попов В. А. Об одном подходе при разработке математического обеспечения автоматизированной системы проектирования дискретных устройств. — Тез. докл. Всесоюз. науч. совещ. посвященной дню радио. — М., 1978, с. 30—31.

Поступила в редакцию 02.02.81.

Н. И. СЕНЧЕНКО, канд. техн. наук, Ю. С. ЗАМАЛЕЕВ АЛГОРИТМ УПРАВЛЕНИЯ l -КРАТНЫМИ КОММУТАТОРАМИ ДЛЯ АНАЛОГОВЫХ ПРОЦЕССОРОВ

При построении систем автоматической коммутации (САК) аналоговых процессоров возникает задача сокращения числа коммутирующих элементов (КЭ). Эту задачу можно решать различными способами [1 — 3]. Один из них состоит в наложении ограничений на сеть связей между вычислительными блоками (ВБ), когда каждый ВБ соединяется не более чем с l другими блоками [4, 5]. Такая коммутация называется l -кратной.

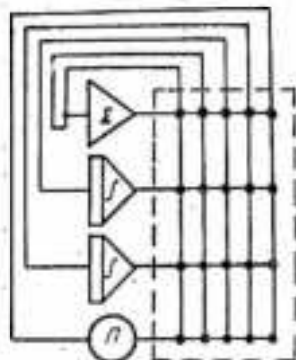


Рис. 1. Прямоугольный матричный коммутатор 5×4 (Σ — сумматор, ∫ — интегратор, П — потенциометр, ● — коммутирующий элемент)

Системы автоматизации программирования (САП) аналоговых процессоров гибридных вычислительных систем (ГВС) составляют произвольные схемы решения, которые не учитывают ограничений l -кратной коммутации. Алгоритмы управления реально существующими коммутаторами являются специальными и предназначены для конкретных САК.

В настоящей работе предлагается универсальный алгоритм упорядочивания схем решения в соответствии с возможностями l -кратной коммутации.

1. Типы коммутаторов аналоговых процессоров.

Для построения САК аналоговых процессоров применяются следующие типы коммутаторов [1, 6]: прямоугольные, двухступенчатые, многоступенчатые.

Самая простая форма коммутатора — прямоугольная, которая позволяет осуществлять необходимые связи без блокировок. Соединение входа и выхода ВБ производится посредством только одного КЭ. Прямоугольный матричный коммутатор с пятью входами и четырьмя выходами требует 20 (5×4) КЭ (рис. 1). При этом общее число КЭ матричного коммутатора $k = u \cdot v$, где u — число входов, а v — число выходов.

Уменьшить количество КЭ можно, используя многоступенчатые коммутаторы. Но применяя электронные КЭ, необходимо учитывать, что их сопротивление в замкнутом состоянии не равно нулю, а в разомкнутом не равно бесконечности. Чтобы не допустить потери точности, число КЭ между входом и выходом ВБ не должно быть более трех [6]. В этом случае не происходит потери точности при моделировании.

Рассмотрим структуру трехступенчатого коммутатора, имеющего u входов и v выходов (рис. 2). Первый каскад содержит x прямо-

угольных ксммутаторов размером $u \times y$, второй — y коммутаторов размером $x \times z$ и третий каскад — z коммутаторов размером $y \cdot v = V$. Такой коммутатор при $u = 64$ и $v = 3$ требует только 768 КЭ, а блокировки в нем практически отсутствуют [7].

Перед установкой соединений к ксммутационной системе предъявляется список требований (список Т). В соответствии с этим списком устанавливаются соединительные пути между входами и выходами ВБ. Если при предъявлении списка Т в системе нет реализованных соединений, то ксммутационная система такого типа называется разовой [8]. В аналоговых процессорах чаще всего применяется разовая ксммутация.

В сетях связи список Т задается в виде матрицы требований, которая для трех-каскадной схемы имеет следующий вид:

$$T = \|t_{ij}\|, \quad (i = 1, 2, \dots, x; j = 1, 2, \dots, z).$$

Здесь элементы t_{ij} являются целыми положительными числами, которые определяют количество необходимых соединений между ксммутатором i входного каскада и ксммутатором j выходного каскада.

В ИЕС список требований на ксммутацию обычно задается в виде графической структурной схемы или посредством ее формального описания [9], где строго определено, какие входы и выходы соединяются между собой. Поэтому задание списка Т на соединение в виде матрицы T для САК аналоговых процессоров неприемлемо.

2. *Представление схем решения в виде условных матриц.* Любую структурную схему можно представить в виде направленного графа. Вершины такого графа будут изображать ВБ, а направленные дуги — указывать последовательность соединения ВБ между собой. Всякий граф может быть описан матрицей инцидентий:

$$C = \|c_{ls}\|, \quad (l = 1, 2, \dots, h; s = 1, 2, \dots, m).$$

Матрица инцидентий схем решения состоит из m столбцов, в которых перечислены ВБ и их номера, и из h строк, в которых перечислены все дуги графа.

Рассмотрим аналоговую структуру A , содержащую ВБ:

$$A = \{a_1^{(\mu_1)}, a_2^{(\mu_2)}, \dots, a_l^{(\mu_l)}, \dots, a_v^{(\mu_v)}\},$$

где v — число типов ВБ; μ_i — число ВБ i -го типа. Тогда, если дуга $w_l = a_l^{(\mu_l)} a_k^{(\mu_k)}$, которая соединяет вершины $a_l^{(\mu_l)}$ и $a_k^{(\mu_k)}$, исходит из вершины $a_l^{(\mu_l)}$ и входит в вершину $a_k^{(\mu_k)}$, то это означает, что ВБ $a_l^{(\mu_l)}$ должен быть соединен с ВБ $a_k^{(\mu_k)}$.

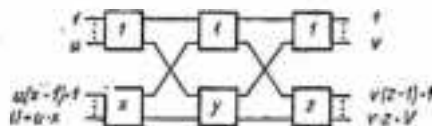


Рис. 2. Трехступенчатый матричный коммутатор

Элементы c_{ls} матрицы инцидентий структурной схемы определяются по следующему правилу:

$$c_{ls} = \begin{cases} +1, & \text{если дуга } \omega_l \text{ исходит от ВБ } a_l^{(s,i)}; \\ 0, & \text{если дуга } \omega_l \text{ не инцидентна ВБ } a_l^{(s,i)}; \\ -1, & \text{если дуга } \omega_l \text{ входит в ВБ } a_l^{(s,i)}. \end{cases}$$

Полная матрица инцидентий для САК содержит большой избыток информации в виде нулей. Поэтому для представления топологии сети связей САК воспользуемся условными матрицами [5]. Последняя представляет собой множество M элементов, определенных следующим образом:

$$m_{lp} = s \cdot c_{ls}; \quad P = 1, 2; \quad m_{lp} \in M, \text{ если } c_{ls} \neq 0; \\ m_{lp} \notin M, \text{ если } c_{ls} = 0.$$

Если задано формальное описание структурной схемы, то построение условной матрицы производится, минуя процесс перехода к матрице инцидентий. При этом любой элемент m_{lp} условной матрицы представляет собой следующее выражение:

$$m_{lp} = (-1)^{(p-1)} (\beta_l + p)_l.$$

Здесь $P = 1, 2$; $l = 1, 2, \dots, v$; β_l — константа, определяемая типом ВБ; l — номер связи между главным и вспомогательным элементами в формальном описании. Алгоритм перехода к условным матрицам представлен в [5].

3. *Алгоритм управления l -кратным коммутатором.* Как уже отмечалось, САП строят произвольные схемы решений, не учитывая возможностей САК. Если аналоговый процессор ГВС использует l -кратный коммутатор, то необходимо упорядочить построенную схему решения, чтобы она соответствовала возможностям САК.

Сформулируем задачу упорядочивания более строго.

Пусть имеется аналоговая структура A , содержащая ВБ, общее число которых $d = \sum_{i=1}^n \mu_i$. Сеть связей между ВБ структуры A определяется коммутирующей матрицей L . Дано множество $B = \{b_1, b_2, \dots, b_n\}$ независимых задач. Любая задача $b_i \in B$, $i = 1, 2, \dots, n$, подлежащая решению на структуре A , может быть разбита на операторы $C = \{c_1^{(p,n)}, c_2^{(p,2)}, \dots, c_n^{(p,n)}\}$. Общее число операторов задачи b_i равно $h_i = \sum_{j=1}^n \mu_{ij}$, $i = 1, 2, \dots, n$. Задача состоит в следующем; можно ли поставить в соответствие каждому оператору $c_k^{(p,n)} \in C$ один из ВБ $a_l^{(p,i)} \in A$ таким образом, чтобы при соединении их в структурную схему, согласно математическому описанию, между ВБ имелись связи.

Для решения этой задачи необходимо выполнение условий:
 1) какую бы группу из h_i ни взяли, должно найтись, по крайней мере, h_i ВБ, каждый из которых может реализовать один из

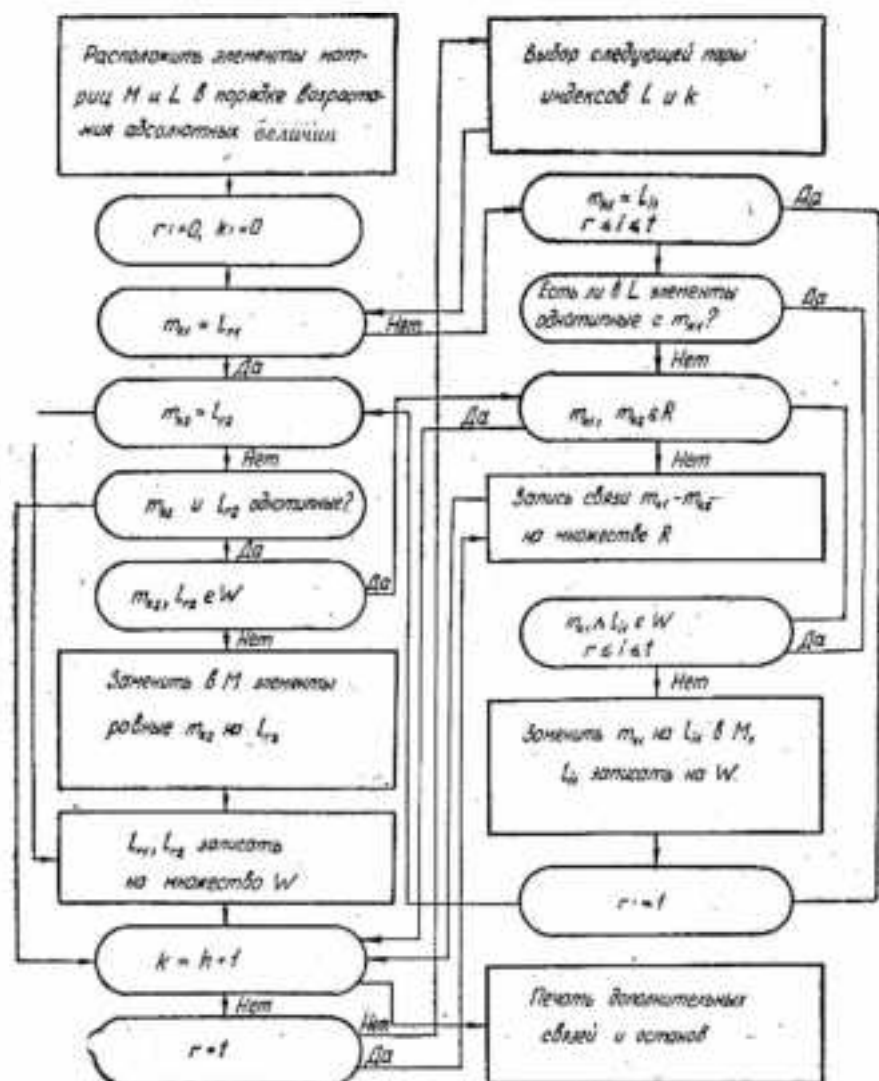


Рис. 3. Алгоритм упорядочивания для i -кратных систем коммутации

операторов. Это условие для краткости назовем условием возможности решения; 2) распределение операторов между ВБ должно соответствовать возможностям коммутирующей матрицы L .

Проверить условие возможности решения весьма просто. Сложнее обстоит дело с проверкой второго условия. Для этого необходимо разработать алгоритм, позволяющий отображать произвольную структурную схему в схему решения, на которую наложены ограничения l -кратной коммутации.

Остановимся на основных операциях алгоритма. Обозначим множество связей между ВВ при полной коммутации через Q . Тогда множество связей R , не вошедших в матрицу L при l -кратной коммутации, образует запрещенные связи $R \cup L = Q$.

Коммутирующая матрица L представляет условную матрицу $L = \|l_{ij}\|$. Следовательно, исходная схема решения, на которую накладываются ограничения l -кратной коммутации, должна быть преобразована в условную матрицу $M = \|m_{kp}\|$. Затем необходимо расположить элементы матриц M и L в порядке возрастания абсолютных величин. После этого отыскать строки, для которых выполняется условие $m_{k1} = l_{11}$, а m_{k2} и l_{12} при этом являются однотипными элементами.

Множество элементов матрицы L l_{11} и l_{12} , посредством которых произвели замену элементов m_{k1} и m_{k2} , обозначим через W . Если m_{k2} и $l_{12} \in W$, то производится замена элементов, равных m_{k2} на l_{12} в матрице M . Если же m_{k2} и $l_{12} \notin W$, то связь $m_{k1} - m_{k2} \in R$. Эта связь выводится на печать. При $m_{k1} \neq l_{11}$ отыскиваем среди элементов $l_{11} \in L$ элементы одного типа с m_{k1} и заменяем m_{k1} и равные ему элементы матрицы M на найденные и не принадлежащие W . В противном случае $m_{k1} - m_{k2} \in R$. Структура алгоритма представлена на рис. 3. Сформулированная таким образом упорядоченная матрица M преобразовывается в формальное описание для управления САК.

Следует заметить, что рассмотренный в данной работе алгоритм может применяться для управления l -кратными коммутаторами цифровых интегрирующих машин и разрядно-аналоговых вычислительных систем, содержащих разнотипные ВВ.

Список литературы: 1. Тараторин Ю. И. Автоматизация набора задач в аналоговой части ГВС (обзор).— Автоматика и телемеханика, 1975, № 2, с. 136—140. 2. Тараторин Ю. И. О синтезе системы автоматизированного набора задач в аналоговой части гибридного вычислительного комплекса.— Автоматика и телемеханика, 1976, № 4, с. 157—165. 3. Сенченко Н. И. Системы автоматической коммутации аналоговых процессоров ГВС.— Электронное моделирование, 1980, № 1, с. 49—55. 4. Мурашко А. Г., Сенченко Н. И., Терентьев М. Ф. Принципы построения системы автоматической коммутации операционных элементов в решающую структуру.— Аналоговая и аналого-цифровая вычислительная техника, 1973, вып. 6, с. 109—114. 5. Красногорова В. С., Мурашко А. Г., Сенченко Н. И. Способ построения l -кратной коммутирующей матрицы для аналоговых структур.— Автоматика и телемеханика, 1974, № 7, с. 172—177. 6. Hecht Mr. V., Reiss Mr. W., Rzehak Mr. H. A hybrid compiler with automatic patching system.— Simulation, 1975, No 6, p. 185—192. 7. Maxwell J. B. A continuous system simulation language for an AHCSSL.— Paper presented at the AICA International Symposium on Simulation Languages for Dynamic Systems. London, 8—10 Sep. 1975, p. 283—293. 8. Харкевич А. Д. Особенности

УДК 681.142.3

А. С. ШКИЛЬ, В. И. ХАХАНОВ

ТРОИЧНОЕ МОДЕЛИРОВАНИЕ ЦИФРОВЫХ УСТРОЙСТВ С ИСПОЛЬЗОВАНИЕМ ЯЗЫКА КУБИЧЕСКИХ КОМПЛЕКСОВ

При логическом моделировании цифрового устройства на ЭВМ его элементы, как правило, описывают системой логических уравнений [1]. Эти уравнения многократно решают для каждого моделируемого входного набора, что существенно снижает скорость моделирования. Поэтому для уменьшения времени моделирования цифровых устройств целесообразно оперировать не с логическими уравнениями, а с их решениями, представленными в виде определенных таблиц.

Одним из широко используемых способов представления решений системы логических уравнений является описание схемы ее вырожденным покрытием [2], т. е. решение системы уравнений записывают на языке кубических комплексов [3]. Вырожденное покрытие цифровой схемы — это совокупность 0-покрытия и 1-покрытия данной схемы. 0-покрытием некоторой цифровой схемы называется множество входных кубов, вызывающих нулевую реакцию на выходе рассматриваемой цифровой схемы. Множество входных кубов, вызывающих на выходе единичную реакцию, называется 1-покрытием схемы.

При описании цифрового устройства в виде логических уравнений моделирование неполностью определенных наборов вызывает существенные затруднения, так как это приводит к необходимости параметрического решения системы уравнений, что значительно усложняет моделирование и увеличивает его продолжительность. Описание цифрового устройства в виде вырожденных покрытий его блоков позволяет эффективно моделировать цифровую схему для неполностью определенных входных наборов (с безразличными символами X на некоторых физических входах рассматриваемой схемы).

Покажем применение данного способа описания блоков цифрового устройства при проведении троичного моделирования цифровых последовательностных схем, которое широко используется при моделировании переходных процессов в цифровых устройствах [4].

Рассмотрим символику, которая используется в предлагаемом ниже методе моделирования. Символ 0 (1) обозначает появление сигнала уровня логического нуля (единицы) в некотором узле цифровой схемы. Символ u обозначает неопределенное состояние некоторого узла цифровой схемы [5]. Физически это соответствует переходному процессу, протекающему в данный момент в некоторой точке цифровой схемы или неопределенному состоянию элемента памяти. Символ X обозначает безразличное состояние некоторого узла цифровой схемы. Символом X помечаются те точки цифровой схемы, значение сигнала в которых не влияет на значение сигнала на выходе схемы. Особенность символа X , в отличие от символа u , состоит в том, что он может быть доопределен любым символом из перечисленных выше. Неопределенное состояние узла схемы, обозначаемое u , не может быть доопределено ни 0, ни 1 и рассматривается как третье состояние некоторого узла цифровой схемы.

Описанная символика используется в тричном моделировании и в описании схем на языке кубических комплексов [3, 5].

Кроме того, введем дополнительные символы Y и Z .

Символ Y означает, что в некотором узле цифровой схемы уровень сигнала равен логической единице или протекает переходный процесс и уровень сигнала не определен, т. е. $Y = 1Ou$, где u — операция объединения.

Символ Z означает, что в некотором узле цифровой схемы уровень сигнала равен логическому нулю или протекает переходный процесс и уровень сигнала не определен, т. е. $Z = 0Ou$. Следовательно, символ X как безразличное состояние узла схемы можно представить в виде $X = 0Ou = YUZ$.

Введем определение u -покрытия.

Определение 1. u -покрытием назовем совокупность входных кубов некоторого цифрового элемента, которые будут приводить к появлению неопределенности на выходе цифрового элемента при условии, что хотя бы на одном из его входов значение сигнала не определено.

u -покрытия логических элементов строятся следующим образом. На одном из входов логического элемента задается символ u . Для того, чтобы на выходе элемента был неопределенный сигнал, на остальных входах этого элемента должны быть или символы u или символы, создающие условие транспортировки [2] и по рассматриваемому входу. По такому принципу строятся кубы u -покрытия по каждому входу.

Используя предложенную символику и правила записи u -покрытий, запишем 0-покрытия, 1-покрытия и u -покрытия для трехходовых логических элементов И, ИЛИ, И — НЕ, ИЛИ — НЕ и инвертора НЕ. Покрытия этих элементов приведены в табл. 1. Для удобства обозначения введем символику, обозначающую кубы покрытия. Некоторый куб покрытия обозначим C_t^i , где t — тип элемента, покрытия которого рассматриваются;

i — тип рассматриваемого покрытия; j — номер по порядку двинного куба в соответствующем покрытии. Использование данной символики показано в табл. 1. Например, куб $C_1^{0,1}$ обозначает, что данный куб является первым кубом μ -покрытия элемента И, т. е. $\mu Y Y u$.

Таблица 1

$C_i^{j,l}$	$C_1^{j,l}$ (И)		$C_2^{j,l}$ (ИЛИ)		$C_3^{j,l}$ (И-НЕ)		$C_4^{j,l}$ (ИЛИ-НЕ)		$C_5^{j,l}$ (НЕ)	
	Вход	Выход	Вход	Выход	Вход	Выход	Вход	Выход	Вход	Выход
$C_1^{0,1}$	0XX X0X XX0	0 0 0	000 0	0	111 0	0	1XX X1X XX1	0 0 0	1	0
$C_1^{1,1}$	111	1	1XX X1X XX1	1 1 1	0XX X0X XX0	1 1 1	000	1	0	1
$C_1^{\mu,1}$	$\mu Y Y$ $Y \mu Y$ $Y Y \mu$	μ μ μ	$\mu Z Z$ $Z \mu Z$ $Z Z \mu$	μ μ μ	$\mu Y Y$ $Y \mu Y$ $Y Y \mu$	μ μ μ	$\mu Z Z$ $Z \mu Z$ $Z Z \mu$	μ μ μ	μ	μ

Отметим, что μ -покрытия совпадают для элементов И и И — НЕ, ИЛИ и ИЛИ — НЕ соответственно в связи с тем, что символа $\bar{\mu}$ не существует [5].

Моделирование цифрового устройства осуществляется путем суперпозиции решений для отдельных блоков в общее решение при заданном входном наборе с использованием операции μ -пересечения. Данная операция является расширением известной операции пересечения кубов [3].

Операцию μ -пересечения двух кубов $a = a_1, a_2, \dots, a_n$ и $b = b_1, b_2, \dots, b_n$ обозначим $c = a \mu b$, где $c = c_1, c_2, \dots, c_n$, и определим следующим образом:

$$c = a \mu b = \begin{cases} \emptyset, & \text{если хотя бы для одного } c_i = a_i \bar{b}_i = \emptyset; \\ (a_1 \bar{b}_1), (a_2 \bar{b}_2), \dots, (a_n \bar{b}_n) & \text{в противном случае.} \end{cases}$$

Таким образом, операция μ -пересечения является покомбинаторной и определяется табл. 2.

Моделирование входного набора для некоторой цифровой схемы осуществляется путем μ -пересечения входного набора с покрытиями последовательно занумерованных элементов. Нумерация элементов схемы осуществляется таким образом, чтобы для каждого элемента номера его выходов были больше номеров его входов. Входной набор сначала пересекается с 0-покрытием, затем с 1-покрытием и затем с μ -покрытием, такой порядок выбора покрытий элемента для пересечения обусловлен необходимостью доопределения неполностью определенных входных на-

боров с целью уменьшения возможности продвижения символов u к внешним выходам схемы.

Современные цифровые устройства содержат схемы различной степени интеграции. Раскрывать структуру схем средней интеграции (СИС) до логических элементов — довольно трудо-

Таблица 2

u	0	1	X	Z	Y	u
0	0	$\emptyset$	0	0	$\emptyset$	$\emptyset$
1	$\emptyset$	1	1	$\emptyset$	1	$\emptyset$
X	0	1	X	Z	Y	u
Z	0	$\emptyset$	Z	Z	u	u
Y	$\emptyset$	1	Y	u	Y	u
u	$\emptyset$	$\emptyset$	u	u	u	u

емкая, а часто и невыполнимая задача. Поэтому целесообразно представить решения уравнений, описывающих логику работы СИС, в виде 0-покрытий, 1-покрытий и u -покрытий для определенного неделимого функционального модуля; 0-покрытия и 1-покрытия для СИС можно получить методом л-алгоритма [3]. Получение u -покрытий для схем средней интеграции связано с расширением классического л-алгоритма на предлагаемую троичную символику.

Для этого введем определение Z -покрытий и Y -покрытий. **Определение 2.** Z -покрытием назовем совокупность входных кубов некоторого цифрового элемента, которые будут приводить к появлению на выходе неопределенного сигнала или сигнала нулевого уровня (символа Z), т. е. на выходе элемента не будет сигнала единичного уровня.

Определение 3. Y -покрытием назовем совокупность входных кубов некоторого цифрового элемента, которые будут приводить к появлению на выходе неопределенного сигнала или сигнала единичного уровня (символ Y), т. е. на выходе элемента не будет сигнала нулевого уровня.

Запишем Z -покрытия и Y -покрытия для элементов, приведенных в табл. 1; Z -покрытия и Y -покрытия этих элементов приведены в табл. 3. Построение Z -покрытий и Y -покрытий аналогично построению u -покрытий. Записав для всех элементов 0 (1) — покрытия, u -покрытия, а также Y -покрытия и Z -покрытия, выполним расширенный л-алгоритм, основой которого является операция u -пересечения. Получение 0-покрытий и 1-покрытий в расширенном л-алгоритме не имеет существенных отличий от классического л-алгоритма. Получение u -покрытий имеет ряд особенностей. На i -м шаге л-алгоритма формируемый куб u -покрытий u -пересекается сначала с Y -покрытием и Z -покрытием рассматриваемого элемента. Если результат этих u -пересечений не равен $\emptyset$, то переходим на следующий шаг алгоритма, если равен $\emptyset$, то u — пересечение затем выполняется с u — покрытием, 1 — покрытием и 0 — покрытием. В дальнейшем выполнение расширенного л-алгоритма не отличается от классического.

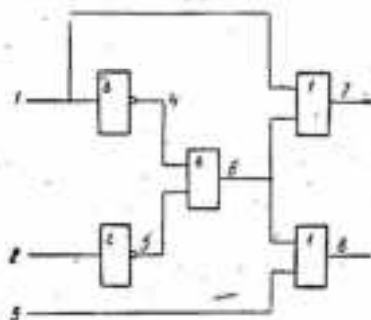
$C_1^{i,j}$	$C_1^{i,j}(и)$		$C_2^{i,j}(или)$		$C_3^{i,j}(и-не)$		$C_4^{i,j}(или-не)$		$C_5^{i,j}(не)$	
	Вход	Выход	Вход	Выход	Вход	Выход	Вход	Выход	Вход	Выход
$C_1^{i,j}$			YXX	Y	ZXX	Y				
	YUY	Y	XUX	Y	XZX	Y	ZZZ	Y	Z	Y
			XXY	Y	XXZ	Y				
$C_2^{i,j}$	ZXX	Z					YXX	Z		
	XZX	Z	ZZZ	Z	YUY	Z	XUX	Z	Y	Z
	XXZ	Z					XXY	Z		

Полученные таким образом μ — покрытия СИС наряду с их 1 — покрытиями и 0 — покрытиями включаются в описанный алгоритм трючного моделирования.

Предложенный алгоритм проиллюстрируем на примере моделирования переходного набора для схемы из [4], приведенной на рисунке. При проведении моделирования в круглых скобках указывается номер элемента в схеме, который моделируется на данном этапе работы алгоритма (табл. 4).

В заключение отметим, что предлагаемый алгоритм трючного моделирования эффективно работает с π -алгоритмом, упоминаемым выше, и D -алгоритмом [2] генерирования диагностических тестов для цифровых последовательностных схем. Кроме того, предлагаемый алгоритм трючного моделирования не требует никакой дополнительной информации, за исключением μ -покрытий, кроме той, что имеется при выполнении D -алгоритма и π -алгоритма.

Программная реализация данного алгоритма подтвердила его эффективность для выполнения трючного моделирования цифровых последовательностных схем.



Пример цифровой схемы

Список литературы: 1. Автоматизация проектирования ЭВМ/Под ред. Л. В. Абрайтиса. — М.: Сов. радио, 1978. — 272 с. 2. Чжен Г., Манкина Е., Метц Г. Диагностика отказов цифровых вычислительных систем. — М.: Мир, 1972. — 232 с. 3. Проектирование цифровых вычислительных систем/Под ред. С. А. Майорова. — М.: Высшая школа, 1972. — 344 с. 4. Eichelberger E. B. Hazard detection in combinational and sequential swit-

M_i	1	2	3	4	5	6	7	8	Пересечения
M_1	u	0	u	X	X	X	X	X	$M_{1u}C_5^{0,1}(4) = \emptyset; \quad M_{1u}C_5^{1,1}(4) = \emptyset; \quad M_{1u}C_5^{u,1}(4) = u.$
M_2	u	0	u	u	X	X	X	X	$M_{2u}C_5^{0,1}(5) = \emptyset; \quad M_{2u}C_5^{1,1}(5) = 1;$
M_3	u	0	u	u	1	X	X	X	$M_{3u}C_1^{0,1}(6) = \emptyset; \quad M_{3u}C_1^{0,2}(6) = \emptyset;$
									$M_{3u}C_1^{1,1}(6) = \emptyset; \quad M_{3u}C_1^{u,1}(6) = u.$
M_4	u	0	u	u	1	u	X	X	$M_{4u}C_2^{0,1}(7) = \emptyset; \quad M_{4u}C_2^{1,1}(7) = \emptyset;$
									$M_{4u}C_2^{1,2}(7) = \emptyset; \quad M_{4u}C_2^{u,1}(7) = u.$
M_5	u	0	u	u	1	u	u	X	$M_{5u}C_2^{0,1}(8) = \emptyset; \quad M_{5u}C_2^{1,1}(8) = \emptyset;$
									$M_{5u}C_2^{1,2}(8) = \emptyset; \quad M_{5u}C_2^{u,1}(8) = u.$
M_6	u	0	u	u	1	u	u	u	

УДК 62—50

В. М. МИХАЙЛЕНКО, д-р техн. наук, А. Н. ЛИТВИНОВ, В. А. ПОЯРКОВ, Г. А. ГОРОШКОВ

МЕТОДОЛОГИЧЕСКИЕ ВОПРОСЫ РАЗРАБОТКИ САПР-СЕТЕЙ КОММУНАЛЬНОГО ХОЗЯЙСТВА

Планируемое развитие больших городов приводит не только к усложнению сетей коммунального хозяйства, но и требует еще на стадии проектирования огромного человеческого труда и больших капитальных вложений, поэтому решение проблемы автоматизации проектирования сетей коммунального хозяйства приобретает важное народнохозяйственное значение.

Сохранение традиционных методов проектирования приводит к возрастанию сроков создания новых и реконструкции существующих сетей водо-газо- и теплоснабжения, что в свою очередь нередко приводит к «моральному износу проектов», т. е. идеи, заложенные в проект, успевают устареть еще до создания сооружений. Необходимость завершения проектов в приемлемые сроки вызывает потребность увеличения численности ИТР, занятых в сфере проектирования и ограничивает число рассматриваемых вариантов проектов, что сказывается на их качестве.

С другой стороны; объем технической документации настолько велик, что для ее оформления расходуется большая часть сил и времени проектировщиков.

Выход из создавшегося положения состоит в создании системы автоматизации проектирования (САПР) с применением ЭВМ, которая в конечном итоге позволяет:

решать многокритериальные задачи анализа и синтеза инженерных сетей с минимальными затратами; выбирать оптимальные режимы работы сетей; обеспечить «живучесть» сетевых систем в аварийных ситуациях при их наращивании; резко сократить сроки выполнения проектных работ; увеличить во много раз точность расчетов, что особенно важно в условиях возрастающего дефицита целевого продукта (вода, газ, тепло и т. д.).

Системное проектирование инженерных сетей базируется на трех уровнях блочно-иерархического проектирования: базовых элементов или микроуровень; фрагментов сетевой системы или макроуровень; сети как комплекса управляемой и управляющей системы или информационный уровень.

Очевидно, что при переходе на более высокий уровень проектирования математическая модель сетевой системы претерпевает существенные изменения. На микроуровне математические модели потребителей, источников и линий связи представляются дифференциальными уравнениями в частных производных. На макроуровне используется представление о сети как о дискретном пространстве, математическая модель представляет собой обыкновенные дифференциальные или алгебраические и трансцендентные уравнения. На этом уровне полная математическая модель сетевой системы получается объединением математических моделей базовых элементов в общие системы уравнений с помощью первого и второго постулатов сетей (аналог — законы Кирхгофа для электрических цепей).

Для построения математической модели третьего уровня используется математическая логика, теория массового обслуживания, методы теории автоматического управления.

Таким образом, с точки зрения кибернетики инженерная сеть — сложная иерархическая система взаимодействия большого количества элементов (подсистем) трех типов: нагрузки (потребители целевого продукта); источники целевого продукта (насосные станции); линии связи (трубопроводы).

Каждому из этих элементов поставлены в соответствие последовательная (расход целевого продукта) и параллельная (потеря напора) переменные величины, характеризующие его состояние и некоторые параметры (диаметры и длины трубопроводов), влияющие на характер зависимости между этими величинами, т. е. состояние всей сети характеризуется двумя векторами q и h и является следствием работы активных источников, принятой схемы соединения базовых элементов и значений их параметров.

Задача проектирования инженерной сети сводится к определению местоположения базовых элементов сети, значений их параметров и переменных состояния, а также схемы соединения этих элементов, обеспечивающих (без нарушения ограничений на параметры и переменные) требуемые потокораспределения в подсистемах потребления при минимальных значениях некоторого критерия (энергозатрат, стоимости и т. д.).

Поскольку САПР является «человеко-машинной» системой проектирования, а в процессе проектирования на определенных этапах отсутствуют конкретные данные о конфигурации сети, о местоположении ее подсистем, их характеристиках и параметрах, то весь процесс проектирования целесообразно разбивать на этапы, на каждом из которых в режиме диалога «человек — машина» необходимо обеспечить рост информативности разработчика и САПР о проектируемой системе вплоть до создания полной модели системы, после чего САПР должна обеспечить изготовление технической документации проектируемой сети.

Назначение и структура программного обеспечения. Математическое обеспечение САПР инженерных сетей реализовано дискретным набором связанных между собой подсистем.

Ведущая программа предназначена для управления всем процессом проектирования, определяет порядок работы всех подсистем, формирует данные, необходимые операционной системе для вызова тех или иных фаз подсистем и передает им управление.

Во время работы САПР в основной программе всегда присутствует корневой сегмент ведущей программы (управляющие таблицы, области связи подсистем, общие подпрограммы, буферы для хранения данных, а также программы распознавания операторов языка управления заданиями). Все остальные сегменты загружаются в основную память СУПЕРВИЗОРОМ по требованию ведущей программы. Программа распознавания, находящаяся в корневом сегменте, определяет по операторам языка управления заданиями вид необходимой работы и подготавливает вызов требуемой программной фазы. После окончания работы управление передается ведущей программе.

Функциональная подсистема реализована в виде набора функциональных блоков анализа и синтеза, оптимизации и коррекции инженерной сети. Она является основной составной частью САПР, для функционирования которой существуют остальные подсистемы. Функциональные модули являются «инструментом» проектировщика, позволяющим ему по имеющимся базам данных, целей и описаний синтезировать сетевую систему, провести анализ потокораспределений и управляемости сети в различных аварийных ситуациях. Оптимизационные задачи могут работать в автономном или пошаговом режиме. В пошаговом режиме ведущую роль играет «проектировщик», который на основании опыта и интуиции диктует условия синтеза и анализа сети.

В автономном режиме приоритет в выборе путей реализации проекта принадлежит функциональной системе САПР. Пользователь задает ей лишь общие цели проектирования и по поступающим соображениям (протоколу работы) ведущей программы следит за ходом анализа и синтеза сети. Оптимальный вариант сети в обоих случаях выбирается проектировщиком.

Подсистемы языков САПР организуют общение пользователя с системой на языке, близком к профессиональному языку пользователя. В состав подсистемы языка описания сетевой системы положена теоретико-множественная концентрация представления графа сетевой системы с учетом параметров конкретных базовых элементов (потребителей, источников, трубопроводов и т. д.).

Интерпретатор языка управления заданием рассчитан на два режима работы функциональной подсистемы — пошаговый и автономный. Первый режим работы рассчитан на квалифицированного пользователя, хорошо представляющего возможности

системы и информационной связи между ее отдельными подсистемами. Во втором (автономном режиме) пользователь, используя операторы управления, формулирует задание на выполнение определенной работы, определяя цель ее без указания конкретных путей их достижения. Пути выполнения задания определяет система на основании имеющихся функциональных модулей и алгоритмов принятия решений.

Подсистема ведения диалога и обучения пользователя позволяет пользователю в режиме диалога создавать новую сеть или реконструировать существующую. Пользователь обращается к системе с помощью директив, которые набираются на дисплее с помощью функциональной клавиатуры, или же используется сетевое меню системы.

В начале сеанса на поле экрана дисплея появляется сообщение.

Укажите шифр исследуемого микрорайона (фрагмента сети). При отсутствии указанного объекта появляется сообщение: *Микрорайон... В архиве отсутствует.*

Если пользователь затрудняется ответить на запрос системы, то она генерирует на экран дисплея порцию информации, позволяющую ему выбрать один из предлагаемых альтернативных ответов:

1. Вывод на дисплей каталога микрорайонов.
2. Вывод на сигд максимального изображения магистральной сети микрорайона.
3. Вывод на сигд минимального изображения магистральной сети микрорайона.
4. Вывод на сигд минимального изображения распределительной сети микрорайона.
5. Вывод на дисплей таблицы режимов работы магистральной сети микрорайона.
6. Вывод на дисплей таблицы режима работы распределительной сети.
7. Вывод на графопостроитель магистральной сети микрорайона.
8. Вывод на графопостроитель распределительной сети микрорайона.
9. Запись результатов в оперативный архив.
10. Вызов следующего микрорайона.
11. Инструкция пользователю.
12. Консультация пользователю.
13. Конец работы.

Пользователь помечает требуемую последовательность директив. При нормальном завершении каждой директивы на дисплей выдается сообщение об этих событиях.

Выполнена директива (имя директивы).

В конце выполнения последовательности директив (или од-

ной директивы) на дисплей выводится меню для следующего этапа.

Выполнение директивы 10 переводит систему в начальное положение и требует ввода шифра необходимого микрорайона. Директива 11 вызывает на экран дисплея инструкцию, объясняющую назначение директив, которые можно ввести с функциональной клавиатуры. Директива 12 вызывает на экран световое меню обучающей системы, которая содержит краткое описание возможности системы и примеры работы с ней.

Подсистема вывода результатов проектирования обеспечивает вывод на графопостроитель графического изображения магистральной (или распределительной) сети требуемого микрорайона. Программные средства, обеспечивающие построение сетевых систем, реализованы на основе метода графического расширения FORTRAN=N, позволяющего за счет расширения синтаксиса языка наиболее компактно описать операции отображения графической информации.

Подсистема управления данными обеспечивает связь рабочих программ функциональной подсистемы, подсистемы вывода и подсистемы обучения пользователя с базами данных, описаний сетевой системы и базой целей проектирования на языке манипулирования данными, ориентированном на теоретико-множественное представление графа сетевой системы.

Одна из основных функций, выполняемых системой автоматизации проектирования сетей, — организация хранения и ведения информации и проектируемой сети. Объем информации о реальной сети требует большой машинной памяти (до сотен мегабайт).

В то же время информация о сети представляет собой многоуровневую иерархическую структуру [3]:

1. Инженерная сеть (водопроводная, газовая и др.)
2. Районы города
3. Микрорайоны
4. Трубопроводы
5. Диаметры
5. Протяженность
5. Сопротивление
5. Типы трубопроводов
5. Начало
6. Тип элемента (источник, потребитель, регулятор)
7. Координаты
7. Параметры
7. Характеристики (расход, напор, производительность)
5. Конец
6. Тип элемента
7. Координаты
7. Параметры
7. Характеристики

Очевидно, что для организации хранения и ведения документации о проектируемой сети необходимо использовать банк данных (БД) — информационную модель синтезируемой сети.

С целью увеличения производительности системы и для удобства работы пользователя информационную базу САПР инженерных сетей удобно представить в виде баз: структуры сети; типовых элементов; диалога и обучения пользователя.

Поскольку БД должен иметь большую емкость и высокую скорость обработки хранимой информации, целесообразно предусмотреть двухуровневую систему хранения информации [1]. На первом (высшем уровне) информация должна храниться на быстродействующих накопителях на магнитных дисках (НМД) при помощи системы управления базами данных (СУБД). На втором (нижнем уровне) хранятся дубли информационной базы, различные версии проектов инженерной сети. Эта информация располагается на накопителях на магнитных лентах (НМЛ) под контролем системы управления архивом САПР (СУА).

С помощью БД представляется возможным обеспечить централизованное хранение информации о сети, независимость информации от использующих ее программ, устранить избыточность хранимой информации. Таким образом, каждая программа может извлечь из БД необходимую структуру данных или же, наоборот, внести изменения в соответствующие базы данных после завершения определенного этапа проектирования.

В базе данных информация о сети представляется многоуровневой иерархической структурой (город, район, фрагмент района, типовый элемент сети), которая отражает топологию инженерной сети, параметры ее элементов и характеристики режимов работы.

База типовых элементов сети включает в себя библиотеку типовых элементов с их параметрами и графическими образами в соответствии с существующими ГОСТами.

База ведения диалога и обучения пользователя содержит информацию о возможностях САПР и предыстории процесса проектирования. В зависимости от уровня специальной подготовки пользователя эта информация извлекается из базы с помощью трех типов запросов [2]: на внутреннем языке САПР (доступны для системного пользователя); жесткие, имеющие фиксированные форматы (доступны для проблемного пользователя), свободные (обеспечивают разовых пользователей справочной информацией).

Рассмотренная модель информационной базы «САПР-сеть» позволяет проектировщику с помощью управляющей подсистемы и функциональных программ успешно решать задачи анализа и синтеза при наращивании и реконструкции инженерных сетей.

Рассмотренные методологические принципы легли в основу разработки теории построения проблемно-ориентированного

комплекса типа «САПР-сеть», выполненной в Киевском инженерно-строительном институте.

Список литературы: 1. Бойков Н. В. Структурная организация одной специализированной системы автоматизированного проектирования. — Управляющие системы и машины, 1977, № 2, с. 78—84. 2. Михайлов Б. М. Автоматизация процесса ведения и изготовления документации программного обеспечения АСУ. — М.: Сов. радио, 1980. — 144 с. 3. Михайленко В. М., Глуценко В. Е. Об одном подходе к проектированию топологии информационных сетей. — АСУ и приборы автоматики, 1979, вып. 51, с. 26—32.

Поступила в редакцию 28.12.81.

УДК 681.325.65

Н. В. АЛИПОВ, канд. техн. наук, Ю. В. ЛЯШЕНКО

ОБ ОДНОМ ПОДХОДЕ К ТРАССИРОВКЕ ДВУХСЛОЙНЫХ ПЕЧАТНЫХ ПЛАТ

Трассировка печатных плат является наиболее трудоемким этапом технического проектирования электронной аппаратуры. Для ускорения этапа проектирования печатных плат в настоящее время создано большое количество различных алгоритмических методов решения задачи трассировки соединений двухслойных печатных плат [1—4].

При проектировании двухслойных печатных плат (ДПП) решаются следующие задачи: минимизация числа межслойных переходов; минимизация суммарной длины перемычек; максимальная автоматизация проектирования ДПП.

В работе [2] предлагается алгоритм, в котором осуществляется трассировка соединений по каналам, выделяемым на плате. Трассировка осуществляется в ортогональных направлениях с закреплением вертикальных и горизонтальных участков трассы за различными слоями платы. Для трасс, которые соединяют контакты различных горизонтальных рядов, в случае если абсциссы их начальной и конечной точек отличаются более, чем на единицу, создается один межслойный переход. Сами трассы при этом разбиваются на вертикальную и горизонтальную составляющие.

В работах [2—4] рассматриваются алгоритмы с обобщенной для первого и второго слоев моделью дискретного рабочего поля (ДРП) платы. В этом случае при трассировке соединений разрешается переход трассы со слоя на слой, используя определенные критерии.

Недостатками изложенных алгоритмов являются: образование в ряде случаев большого количества межслойных переходов; ручная доводка результатов трассировки; возникновение не разведенных соединений. Следует заметить, что не разведенные трассы реализуются введением перемычек между контактами,

Описан один из алгоритмов машинного конструирования соединений в ДПП. Особенностью предлагаемого алгоритма является возможность исключения ручной доводки результатов трассировки за счет получения путевых координат всех соединений. Для получения координат алгоритм создает совмещенную топологию схемы, организуя прокладку трасс с учетом минимизации числа слоев платы, образующих в результате их проведения. Используя полученные координаты трасс, можно создавать перемычки не между соединяемыми контактами, а только над участками ДПП, в которых невозможно организовать межслойный переход. В результате получаются перемычки минимальной длины.

Трассировка соединений осуществляется в три этапа:

- 1) создание совмещенной топологии схемы, которой ставится в соответствие граф $G = (X, U)$. Вершины графа $x \in X$ соответствуют отдельным проводникам, а ребра $u \in U$ — их пересечения;
- 2) выделение из графа G максимального бихроматического подграфа $G_1 = (X_1, U_1)$, ($x_1 \in X_1$ — проводники, разведенные на двух слоях платы без пересечений, $u_1 \in U_1$ — их пересечения);
- 3) размещение на плате проводников, входящих в граф $G_2 = (X_2, U_2)$, ($X_2 = X \setminus X_1$, $x_2 \in X_2$ — проводники, размещаемые на плате с помощью межслойных переходов и перемычек, $u_2 \in U_2$ — их пересечения).

Получение совмещенной топологии трасс межсоединений осуществляется с помощью модифицированного волнового алгоритма. В качестве основного критерия при построении трасс выбирается число слоев, образующихся при проведении данной трассы. В этом заключается главное отличие алгоритма от описанных, которые минимизируют не число слоев, а геометрические параметры соединений (длину, количество изгибов, число пересечений и т. п.). Использование числа слоев в качестве критерия трассировки позволяет минимизировать хроматическое число графа G . Благодаря этому минимизируется число проводников, не разводимых на двух слоях без межслойных переходов, а получаемые платы имеют незначительное количество межслойных переходов.

Модификация волнового алгоритма заключается в использовании при формировании фронта волны не одного, а двух индексов: длины и слоя. Формирование первого индекса происходит как в обычном алгоритме. Второму индексу присваивается значение, равное числу слоев платы, образующемуся при проведении трассы через данную ячейку ДПП. В процессе работы алгоритма образуются два фронта волны. Первый фронт идентичен фронту волны в обычном алгоритме, а второй — хранит координаты ячеек, в которых произошло увеличение индекса слоя. В случае, если первый фронт не может достигнуть конечной точки трассы, волна начинает формироваться из ячеек второго фронта, имеющих минимальный индекс слоя.

Для определения числа слоев платы разработан быстродействующий алгоритм нахождения хроматического числа графа, сущность которого состоит в следующем.

Пусть на плате проведено n трасс, пересечения между которыми отобразим графом $G_3 = (Y, V)$ (вершины $y \in Y$ соответствуют отдельным трассам, а ребра $v \in V$ их пересечениям). Для каждой вершины $y_i (i = \overline{1, n})$ полученного графа построим подмножества смежности K_i , включая в него вершины, смежные с вершиной y_i . Не смежные вершины графа G_3 y_i и y_j объединим в одну вершину y_{ij} . При объединении вершин могут возникать исходы: а) $K_i \cap K_j = K_i, K_j$; б) $K_i \cap K_j = K_i \cup K_j \cap K_j = K_j$; в) $K_i \cap K_j \neq K_i \cap K_j \cap K_j \neq K_i \cap K_j \cap K_j \neq \emptyset$; г) $y_i \wedge y_j$ принадлежат различным связанным компонентам графа.

Показано, что в случае исходов а), б), г), количество циклов нечетной длины не возрастает, а следовательно, не возрастает оценка хроматического числа графа G_3 . При исходе в) количество циклов нечетной длины изменится, а следовательно, может возрасти оценка хроматического числа графа. Для уменьшения этой возможности из различных вариантов соединения, приводящих к исходу в), выбирается тот, для которого выполняется условие:

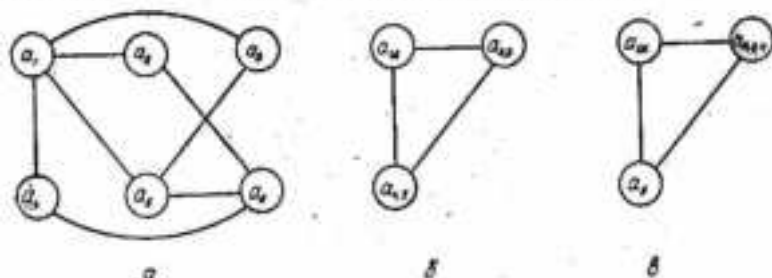
$$|K_i| + |K_j| - 2|K_i \cap K_j| = \min.$$

Процесс объединения продолжается до получения полного графа, количество вершин которого будет оценкой хроматического числа графа G_3 . Разработанный алгоритм хотя и не во всех случаях точно определяет минимальное хроматическое число графа, но позволяет в большинстве случаев избежать принятия эвристических решений и значительно ускоряет процесс определения числа слоев платы.

После сформирования совмещенной топологии схемы с помощью алгоритма определения хроматического числа графа осуществляется раскраска вершин графа G . Затем выбираются два цвета, которые окрашивают максимальное количество вершин x . Вершины, окрашенные в эти цвета, присваиваются графу G_1 . Однако среди оставшихся вершин графа G могут быть вершины, которые не имеют смежностей с объединенными вершинами графа G_1 , соответствующих двум слоям платы.

Рассмотрим описанный случай на примере. Пусть на плате проведено шесть трасс, пересечения которых отображаются графом, приведенным на рисунке (а). В результате работы алгоритма определения хроматического числа графа получается полный граф (рисунок, б). Допустим, что вершины $a_{1,6}$ и $a_{2,3}$ отнесены к первому и второму слоям соответственно. Однако видно, что вершина a_4 , объединенная с вершиной a_5 в $a_{4,5}$, не имеет смежностей с вершиной $a_{2,3}$. Поэтому вершину a_4 объединяем с вершиной $a_{2,3}$ получаем граф, приведенный на рисунке (в).

Для выявления подобных вершин в графе G рассматривается граф, состоящий из трех вершин, две из которых являются объединенными вершинами графа G_1 , третья соответствует вершине в графе G . Процедура определения хроматического числа графа применяется последовательно ко всем вершинам графа G , не вошедшим в граф G_1 . Вершины, не имеющие смежностей с одной из объединенных вершин графа G_1 , соединяются с вершинами графа G_1 , остальные с вершинами графа G_2 .



Выделение бихроматического подграфа

Таким образом, перед началом третьего этапа трассировки имеются рисунки первого и второго слоев платы, на которых размещается основное количество разведенных проводников. Трассировка не разведенных соединений x_2 осуществляется по их путевым координатам с последовательным переходом со слоя на слой. Перед началом разводки очередной трассы x_{21} осуществляется анализ массива смежностей K_1 , позволяющий выбрать такое расположение первого фрагмента трассы на один из слоев, при котором значительно уменьшается количество межслойных переходов. В случае, если трасса встречает препятствие, не преодолимое с помощью межслойного перехода, организуется перемычка на путевых координатах трассы. Перемычка выполняется непосредственно над сложными участками ДРП и поэтому имеет минимальную длину. После разводки всех соединений x_2 на печать выдаются координаты построенных трасс, межслойных переходов и перемычек, а также рисунки слоев платы.

Изложенный алгоритм еще на этапе трассировки позволяет получить совмещенную топологию схемы, удобную для реализации ее на двух слоях платы. Необходимо отметить, что он дает возможность осуществить 100%-ую разводку трасс соединений и исключает применение ручного труда на этапе технического проектирования ДПП.

Список литературы: 1. Селютин В. А. Машинное конструирование электронных устройств.— М.: Сов. радио, 1977.— 384 с. 2. Абрайтис Л. В., Шейнаускас Р. И., Жиливичус В. А. Автоматизация проектирования ЭВМ.— М.: Сов. радио, 1978.— 272 с. 3. Орловский Г. В. Система алгоритмов для трассировки

двухсторонних и многослойных печатных плат.— Обмен опытом в радиопромышленности, 1975, вып. 6, с. 107—112. 4. Федюлов А. А., Битюцкий В. Е., Тортаншина Н. А. Алгоритм трассировки печатных соединений.— Обмен опытом в радиопромышленности, 1975, вып. 6, с. 112—115.

Поступила в редколлегию 14.11.80.

УДК 681.3

А. Х. ГАНИТУЛИН, канд. техн. наук. В. В. МАЗАНИК

МОДЕЛЬ ДЛЯ ОЦЕНКИ ЭФФЕКТИВНОСТИ ФУНКЦИОНИРОВАНИЯ РАСПРЕДЕЛЕННОЙ ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЫ

Для оперативной обработки данных в режиме реального времени широко используются распределенные вычислительные системы (РВС). Создание и применение РВС приводят к необходимости разработки математических моделей, обеспечивающих исследование процессов распределенной обработки данных с целью выбора на этапе проектирования рациональных характеристик РВС и оценки ожидаемой эффективности от их внедрения в конкретных системах управления [1].

В статье излагается имитационная математическая модель РВС, представляющая собой комплекс территориально-распределенных вычислительных средств (ВС) и сеть передачи данных (СПД). ВС состоят из однопроцессорных или многопроцессорных ЭВМ и реализуют частные алгоритмы (ЧА), в соответствии с которыми осуществляется обработка входного потока заявок [2]. СПД обеспечивает информационное взаимодействие между ВС и состоит из линий связи и центров коммутации, обеспечивающих соединение линий связи при обмене данными между ВС.

Управление вычислительным процессом в РВС обеспечивается двухуровневой операционной системой (ОС), состоящей из ОС верхнего уровня — ОС РВС и нижнего уровня — ОС ВС; ОС ВС работает под управлением ОС РВС.

В РВС в случайные моменты времени поступают заявки на обслуживание. Каждая заявка характеризуется своим типом и проходит несколько этапов обслуживания. ОС РВС планирует очередность обслуживания заявок и назначает их на выполнение на ВС с учетом времени обмена данными между ВС. ОС ВС планирует очередность выполнения ЧА, обеспечивающих обслуживание заявок, и назначает их на процессоры ВС.

Моделирование функционирования такой РВС должно осуществляться с учетом характеристик потоков заявок на реализацию ЧА, характеристик алгоритмов, особенностей структуры и организации управления вычислительным процессом в РВС, характеристик ВС.

При моделировании и оценке эффективности РВС учитываются следующие характеристики потоков заявок: номер заявки k , тип k -й заявки h_k , число этапов обработки k -й заявки Φ_k , вид закона распределения промежутков времени между заявками h -го типа f_h , его математическое ожидание M_h и среднеквадратическое отклонение S_h , длина передаваемого сообщения R_h заявки h -го типа. Совокупность перечисленных характеристик задается вектором характеристик заявки: $Z_k = (k, h_k, \Phi_k, f_h, M_h, S_h, R_h)$.

Каждая заявка, поступившая в РВС, обслуживается подмножеством ЧА, определяемым ее типом. Это подмножество характеризуется вектором характеристик алгоритмов $A(h) = (P(h), T(h), \Sigma(h), F(h), Q(h))$, где $P(h) = \{P_i\} (i = \overline{1, I_h})$ — подмножество ЧА, обслуживающих заявку h -го типа (I_h — число ЧА для обслуживания заявки h -го типа); $T(h) = \{T_i\} (i = \overline{1, I_h})$ — множество средних времен реализации ЧА при обслуживании заявки h -го типа; $\Sigma(h) = \{\Sigma_i\} (i = \overline{1, I_h})$ — множество среднеквадратических отклонений времен реализации ЧА при обслуживании заявки h -го типа, $F(h)$ — вид закона распределения времен реализации ЧА при обслуживании заявки h -го типа; $Q(h) = (Q_{\mu, \nu}) (\mu, \nu = \overline{1, I_h})$ — треугольная матрица следования [3] ЧА при обслуживании заявки h -го типа ($Q_{\mu, \mu} = 1$, если μ -й ЧА зависит от ν -го, $Q_{\mu, \nu} = 0$ в противном случае).

Структурные характеристики РВС задаются вектором ST структурных характеристик $ST = (Y, M, c, D)$, где Y — число территориально-рассредоточенных ВС РВС; $M = (m_{pq}) (pq = \overline{1, Y})$ — матрица пропускных способностей СПД ($m_{p,q}$ — пропускная способность линии связи между p -м и q -м ВС); c — параметр, характеризующий метод коммутации СПД ($c = 1$ — коммутация каналов, $c = 2$ — пакетов, $c = 3$ — сообщений [1]), D — множество маршрутов передачи данных в СПД.

Для учета организации вычислительного процесса в РВС используются векторы $E^{РВС}$ и $E^{ВС}$ характеристик ОС РВС и ОС ВС соответственно; $E^{РВС} = (F^{РВС} T^{РВС} Q^{РВС} R^{РВС}, r)$, $E^{ВС} = (F^{ВС} T^{ВС} Q^{ВС} R^{ВС})$, где $F^{РВС} = \{f_{\xi}\} (\xi = \overline{1, \eta})$, $F^{ВС} = \{f_z\} (z = \overline{1, \varepsilon})$ — множество функций, выполняемых программами ОС РВС и ОС ВС соответственно; $T^{РВС} = \{t_{\xi}\} (\xi = \overline{1, \eta})$, $T^{ВС} = \{t_z\} (z = \overline{1, \varepsilon})$ — множество средних времен выполнения каждой программы, реализующей соответствующую функцию ОС РВС и ОС ВС соответственно; $Q^{РВС} = \{q_{\xi}\} (\xi = \overline{1, \eta})$, $Q^{ВС} = \{q_z\} (z = \overline{1, \varepsilon})$ — множество условий обращения к каждой программе, реализующей соответствующую функцию ОС РВС и ОС ВС соответственно; $R^{РВС} = \{p_{\xi}\} (\xi = \overline{1, \eta})$, $R^{ВС} = \{p_z\} (z = \overline{1, \varepsilon})$ — множество вероятностей обращения к каждой программе, реализующей соответствующую функцию ОС РВС и ОС ВС соответственно, r — параметр, характеризующий тип управления ОС РВС ($r = 0$ — децентрализованное, $r = 1$ — централизованное).

Характеристики ВС в модели учитывают с помощью вектора характеристик ВС $\Delta_j = (m_j, p_j)$; $\Delta = \{\Delta_j\} (j = 1, \dots, Y)$, где m_j — число процессоров j -го ВС; p_j — производительность одного процессора j -го ВС.

Совокупность векторов $Z_k, A(h), ST, E^{PBC}, E^{BC}, \Delta$ обеспечивает формализованное задание характеристик потоков заявок, ЧА, особенностей структуры и организации управления вычислительным процессом, характеристик ВС.

В основу моделирующего алгоритма положен событийный принцип. Процессы распределенной обработки данных в РВС по функциональному назначению образуют следующие классы: поступление заявок на обслуживание; планирование ОС РВС прохождения заявок в РВС; назначение ОС РВС заявок для обслуживания на ВС; обмен данными в РВС; планирование выполнения ЧА на ВС; обмен данными в РВС; планирование ОС ВС очередности выполнения ЧА на ВС; диспетчеризация ОС ВС вычислений на ВС; выполнение одного ЧА на процессоре.

Рис. 1. Схема моделирующего алгоритма: 1 — формирование списка событий; 2 — управление обработкой событий; 3 — поступление заявок на обслуживание; 4 — планирование прохождения заявок в РВС; 5 — назначение заявок на ВС; 6 — обмен данными в РВС; 7 — планирование выполнения ЧА на ВС; 8 — назначение ЧА на процессоры; 9 — выполнение ЧА на процессоре; 10 — определение показателей эффективности

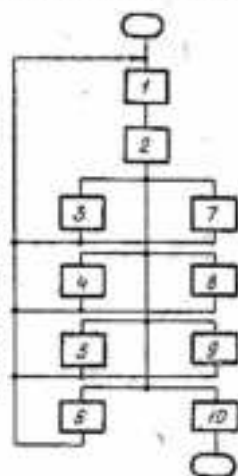


Схема моделирующего алгоритма представлена на рис. 1. Модель функционирует следующим образом. Формируется список событий (процедура 1), в который под управлением блока управления обработкой событий (процедура 2) заносится входной поток заявок (процедура 3). Из списка событий выбирается k -я заявка и в соответствии с ее типом h_k и временем поступления t_k ставится в очередь (процедура 4).

Затем k -я заявка из очереди с учетом времени обмена данными назначается на выполнение на j -е ВС (процедура 5). Данные по назначенной заявке передаются на j -е ВС за время $t_{об}$ (процедура 6). Независимые ЧА (для которых строки в матрице следования содержат все нулевые элементы) для обслуживания k -й заявки строятся в очередь (процедура 7) и поочередно назначаются на свободные процессоры (процедура 8). i -й ЧА реализуется на процессоре j -го ВС за время t_{ij}^0 (процедура 9). Из матрицы следования $Q(h) = (Q_{ij})$, соответствующей типу k -й заявки, вычеркивается столбец $j=i$ (процедура 8), определяются независимые ЧА для обслуживания k -й заявки, строятся в очередь (процедура 7) и поочередно назначаются на свободные про-

процессоры (процедура 8, t_{ij}^{BC} — суммарное время планирования и диспетчеризации ОС j -го ВС при реализации i -го ЧА). Обслуживание k -й заявки на j -м ВС заканчивается, когда в матрице следования $Q(h)$ все элементы Q_{jk} будут равны нулю; k -я заявка вычеркивается из очереди и на j -е ВС назначается следующая заявка (процедура 5). После обслуживания всех заявок из очереди одна реализация модели заканчивается.

Модель позволяет определить количественные значения следующих показателей эффективности:

интегральная пропускная способность

$$z(0, t) = \frac{1}{N} \sum_{\gamma=1}^N \frac{n_{\gamma}^0(0, t)}{n_{\gamma}^a(0, t)},$$

где $n_{\gamma}^0(0, t)$ — число обслуженных РВС заявок на отрезке $(0, t)$ при γ -й реализации моделируемого процесса; $n_{\gamma}^a(0, t)$ — число поступивших в РВС заявок на отрезке $(0, t)$ при γ -й реализации моделируемого процесса; N — количество реализаций моделируемого процесса, обеспечивающих требуемую надежность и точность результатов моделирования;

среднее время обслуживания заявки

$$T = \frac{1}{N \cdot K} \sum_{\gamma=1}^N \sum_{k=1}^K \sum_{j=1}^J \left(\left(\sum_{i=1}^I t_{ijk1}^{CH} + t_{ijk1}^{BC} \right) + t_{ijk1}^{PBC} + \sum_{l=1}^J t_{ijk1}^{обм} \right),$$

где t_{ijk1}^{CH} — время реализации i -го ЧА на j -м ВС при обслуживании k -й заявки в γ -й реализации; t_{ijk1}^{BC} , t_{ijk1}^{PBC} — время работы соответственно ОС ВС и ОС РВС на j -м ВС при обслуживании k -й заявки в γ -й реализации; $t_{ijk1}^{обм}$ — время обмена между j -м и l -м ВС при обслуживании k -й заявки в γ -й реализации; I — число ЧА, реализуемых в РВС;

коэффициент загрузки ВС $K_3 = \frac{1}{N} \sum_{\gamma=1}^N \frac{T_{\gamma}}{t_{\gamma} \cdot L}$, где T_{γ} — время,

затрачиваемое ВС на обслуживание заявок в γ -й реализации; t_{γ} — время γ -й реализации моделируемого процесса; L — суммарное число процессоров в РВС.

Рассмотрим расчет выбранных показателей эффективности при одной реализации моделируемого процесса. В примере рассматривается РВС, состоящая из двух ВС, каждое из которых вклю-

чает два процессора. В случайные моменты времени t_1 и t_2 соответственно, на вход РЭС поступают на обслуживание первая и вторая заявки различных типов (рис. 2). Первая заявка вызывает к реализации 3 ЧА, 1, 2 и вторая — 4, 5 и 6. Векторы характеристик заявок имеют вид:

$$Z_1 = (1, 1_1, 1_1, f_1 = \lambda e^{-\lambda t}, \frac{1}{\lambda}, \frac{1}{\lambda}, 120);$$

$$Z_2 = (2, 2_2, 1_2, f_2 = \lambda e^{-\lambda t}, \frac{1}{\lambda}, \frac{1}{\lambda}, 120).$$

Вектор характеристик алгоритмов

$$A(1) = (P(1), T(1), \Sigma(1), F(1), Q(1)); A(2) = (P(2), T(2), \Sigma(2), F(2), Q(2));$$

$$Q(1) = \begin{array}{c|c|c} 1 & & \\ \hline 2 & 1 & \\ \hline 3 & 1 & 0 \end{array} \quad Q(2) = \begin{array}{c|c|c} 4 & & \\ \hline 5 & 1 & \\ \hline 6 & 1 & 0 \end{array} \quad \begin{array}{l} P(1) = (1, 2, 3); P(2) = \\ = (4, 5, 6); T(1) = (t_1, t_2, t_3); \\ T(2) = (t_4, t_5, t_6); \Sigma(1) = \\ = \Sigma(2) = (0, 0, 0). \end{array}$$

Вектор структурных характеристик

$$ST = (2, M, 1, 1); M = \begin{array}{c|c} 1 & 2 \\ \hline 0 & m_{21} \\ \hline 2 & m_{12} & 0 \end{array}; m_{12} = m_{21} = m = 1200.$$

Векторы характеристик ОС РЭС и ОС ВС соответственно: $E^{РЭС} = (F^{РЭС}, T^{РЭС}, Q^{РЭС}, P^{РЭС}, 1); E^{ВС} = (F^{ВС}, T^{ВС}, Q^{ВС}, P^{ВС})$ где

$$F^{РЭС} = (1, 2), T^{РЭС} = (t_1^{РЭС}, t_2^{РЭС}); t^{РЭС} = \frac{1}{3} t_1^{РЭС} + \frac{2}{3} t_2^{РЭС} = 0,01,$$

$Q^{РЭС} = \{\text{при поступлении очередной заявки; после окончания планирования (первой функции), после освобождения ВС по окончании обслуживания на нем заявки}\};$

$$P^{РЭС} = \left\{ \frac{1}{3}, \frac{2}{3} \right\}; F^{ВС} = (1, 2); T^{ВС} = (t_1^{ВС}, t_2^{ВС});$$

$$t^{ВС} = \frac{1}{3} t_1^{ВС} + \frac{2}{3} t_2^{ВС} = 0,01;$$

$Q^{ПЭ} = \{\text{при появлении независимого ЧА, после окончания планирования (первой функции), после освобождения процессора}\};$

$P^{ВС} = \left\{ \frac{1}{3}, \frac{2}{3} \right\}$. Вектор характеристик ВС:

$$\Lambda_1 = (2, 1); \Lambda_2 = (2, 1); t_{11}^{ПЭ} = \dots = t_{52}^{ПЭ} = 1; t_i^{РЭС} = t_j^{РЭС} = 0,01;$$

$$t_{ij}^{ПЭ} = R/m = 0,1; t_i^{ВС} = t_{ij}^{ВС} = 0,01.$$

Временная диаграмма работы РЭС для рассматриваемого примера представлена на рис. 2. Подставив исходные данные в выражения для выбранных показателей эффективности получим $\delta(0, t_1) = 2/2 = 1$; $T = 3,14$;

$$K_3 = \frac{2(T - t_{\text{обм}})}{4(t_{11}^{\text{вс}} + t_{11}^{\text{вс}} + t_{11}^{\text{ср}} + t_{21}^{\text{вс}} + t_{21}^{\text{ср}} + t_{22}^{\text{ср}} + t_{21}^{\text{обм}})} = \frac{6,68}{4 \cdot 3,13} \approx 0,486.$$

Предлагаемая модель позволяет осуществлять оценку эффективности реализации алгоритмов на РВС с учетом характеристик потоков заявок, алгоритмов, особенностей структуры и организации управления вычислительным процессом в РВС, характеристик ВС; обосновывать требования к структурным характеристикам РВС; выбирать способы организации управления вычислительным процессом, обеспечивающие эффективное обслуживание поступивших заявок.

Список литературы: 1. Пятибратов А. П. Вычислительные системы с дистанционным доступом. — М.: Энергия, 1979. — 191 с. 2. Акимов Л. П. Реализация комплексных алгоритмов сложных систем управления в реальном масштабе времени. — Техническая кибернетика, 1974, № 3, с. 144—149. 3. Барский А. Б. Планирование параллельных вычислительных процессов. — М.: Машиностроение, 1980. — 191 с.

Поступила в редакцию 26.06.81.

УДК 621.313.36

Н. И. ЖУК, канд. техн. наук, В. Д. ОСТРОУХОВ,
канд. техн. наук

К ВОПРОСУ ОПРЕДЕЛЕНИЯ ОПТИМАЛЬНЫХ ПАРАМЕТРОВ СИСТЕМЫ АВТОМАТИЧЕСКОЙ ОЧИСТКИ ТЕХНОЛОГИЧЕСКОГО СЫРЬЯ

Качество готовой продукции и работоспособность технологического оборудования во многих областях производства в решающей степени зависят от чистоты исходного сырья. В условиях современного высокомеханизированного производства достаточно высока вероятность попадания в сырье различного вида металлических частиц, для обнаружения и удаления которых разрабатываются специальные устройства. Эти устройства должны обладать высокой чувствительностью к различным металлам и необходимой устойчивостью в широком диапазоне изменения внешних условий. Реализация системы с указанными требованиями возможна при условии тщательного учета всех факторов, определяющих информационные признаки сигналов. Для этого необходимо знать характер распределения возбуждающего поля в рабочей области пространства, определить диапазон частот, чувствительность приемной аппаратуры, мощность передающей, параметры излучателя, взаимное расположение приемных и

излучающих рамочных антенн с учетом конфигурации металлических включений в потоке исходного сырья.

Чувствительность приемного устройства системы обнаружения и удаления металлических включений в исходном сырье определяется величиной ЭДС полезного сигнала E , наводимого в приемной рамке датчика. В свою очередь, ЭДС сигнала зависит от напряженности вторичного магнитного поля, возбуждаемого металлической частицей. Вторичное магнитное поле запаздывает по отношению к первичному возбуждающему полю, поэтому информационными признаками являются интенсивность вторичного поля и его фаза. Модуль и фаза вторичного поля зависят от формы, размеров и материала частицы, а также от частоты возбуждающего поля. Рассмотрим эту зависимость. Учитывать все возможное многообразие форм металлических частиц не обязательно, так как основные закономерности поведения вторичного поля в зависимости от рассматриваемых параметров можно проиллюстрировать на примере частиц сферической и цилиндрической формы.

Пусть частица сферической формы радиусом a , обладающая магнитной проницаемостью μ и удельной электрической проводимостью σ помещена в непроводящую среду с магнитной проницаемостью μ_0 , в которой действует первичное однородное гармонически изменяющееся поле $H_z e^{i\omega t}$. Требуется определить напряженность возбуждаемого частицей вторичного магнитного поля и характер распределения этого поля в пространстве, окружающем эту частицу. Решение поставленной задачи выполняется традиционными методами*. Выражение для составляющих напряженности магнитного поля получается в следующем виде

$$H_r = H_z \left(1 + \frac{D}{\eta}\right) \cos \vartheta; \quad H_\vartheta = -H_z \left(1 - \frac{D}{\eta}\right) \sin \vartheta, \quad (1)$$

где $\eta = \frac{r}{a}$ — относительное удаление точки наблюдения от центра сферической частицы; ϑ, r, φ — координаты сферической системы; D — аномальная функция, характеризующая вторичное поле. Причем, аномальная функция определяется следующим выражением:

$$D = - \frac{(2\mu + \mu_0) \cdot \eta \cdot I_{-\frac{1}{2}}(\eta) - [\mu_0(1 + \eta^2) + 2\mu] \cdot I_{\frac{1}{2}}(\eta)}{(\mu - \mu_0) \cdot \eta \cdot I_{-\frac{1}{2}}(\eta) + [\mu_0(1 + \eta^2) - \mu] \cdot I_{\frac{1}{2}}(\eta)}, \quad (2)$$

где $\eta = \sqrt{j\omega\mu a}$, a — обобщенный параметр; $I_{\frac{1}{2}}(\eta)$, $I_{-\frac{1}{2}}(\eta)$ — функции Бесселя полуцелого порядка от комплексного аргумента.

* Сметов Б. С. Теория, методика и интерпретация материалов низкочастотной индуктивной электроразведки. — М.: Недра, 1973. — 254 с.

Функция D , определяющая вторичное магнитное поле на границе раздела сфера — вмещающая среда, сложным образом зависит от параметров частицы (μ , σ , a) и от частоты возбуждающего поля ω . Задаваясь размерами частиц, подлежащих обнаружению, можно определить характер частотных зависимостей модуля и фазы вторичного поля от различных материалов. График частотных зависимостей модуля (D) и угла сдвига фазы φ для частиц из железа ($\mu = 10^3 \mu_0$, $\sigma = 10^7 \frac{\text{См}}{\text{см}}$ и $\mu = 10^2 \mu_0$, $\sigma = 10^7 \frac{\text{См}}{\text{см}}$), нержавеющей стали ($\mu = 2,5 \mu_0$, $\sigma = 10^7 \frac{\text{См}}{\text{см}}$), бронзы ($\mu = \mu_0$, $\sigma = 0,55 \times 10^7 \frac{\text{См}}{\text{см}}$), меди ($\mu = \mu_0$, $\sigma = 5,8 \cdot 10^7 \frac{\text{См}}{\text{см}}$) представлены на рис. 1. Радиус сферической частицы a принят равным 1 мм.

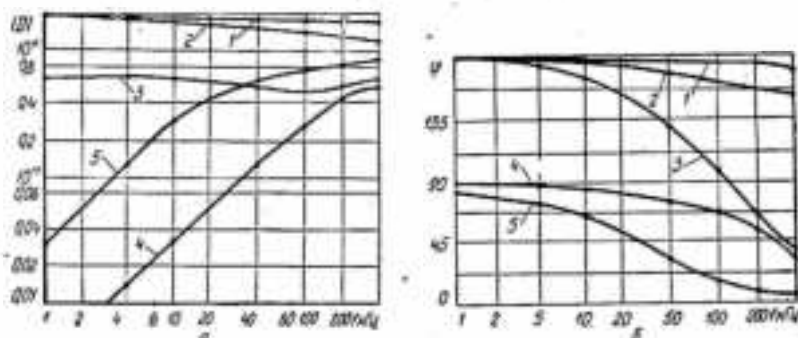


Рис. 1. Амплитудно-а) и фазочастотные б) характеристики вторичных магнитных полей:

1, 2 — железо: $\mu = 10^3$, $\mu = 10^2$; 3 — нержавеющая сталь; 4 — бронза; 5 — медь

Из рассмотрения зависимостей $|D| = F(f)$ (рис. 1, а) следует, что модуль аномальной функции для частиц из ферромагнитных материалов в широком диапазоне частот изменяется незначительно. При уменьшении частоты $|D| \rightarrow 2$; при увеличении частоты наблюдается тенденция к уменьшению $|D|$. Значение $|D|$ для частицы из нержавеющей стали в широком диапазоне частот ($f = 1 \div 400$ кГц) изменяется около среднего значения $|D| = 0,64$ не более, чем на $\pm 5\%$. Для немагнитных металлов (медь: бронза) вторичное магнитное поле существует только за счет действия вихревых токов. Поэтому с уменьшением частоты вторичное поле уменьшается. По мере увеличения частоты вторичное поле растет, в пределе приближаясь к значениям, близким к 1.

Угол сдвига фаз между первичным и вторичным полями для ферромагнитных материалов в рассматриваемом диапазоне частот изменяется от 145 до 180° , для нержавеющей стали $\varphi = 40$ — 180° , а для немагнитных металлов $\varphi = 0$ — 90° (рис. 1, б). Из

рассмотрения частотных характеристик вторичных полей следует, что при использовании в качестве информационных признаков о наличии металлической частицы в потоке сырья интенсивности и фазы вторичного поля, рабочую частоту возбуждающего поля целесообразно выбирать в диапазоне 50—150 кГц.

Достаточно типичным является также случай, когда инородная металлическая частица имеет цилиндрическую форму. Аномальная функция, описывающая вторичное поле на поверхности цилиндра, выражается через цилиндрические функции первого рода второго и нулевого порядков от комплексного аргумента:

$$T = \frac{I_2(\gamma)}{I_0(\gamma)}. \quad (3)$$

Магнитное поле во внешней среде по отношению к цилиндрическому току изменяется в соответствии с выражениями:

$$H_z = H_{z0} \left(1 - \frac{T}{\gamma}\right) \cos \varphi; \quad H_\varphi = -H_{z0} \left(1 + \frac{T}{\gamma}\right) \sin \varphi, \quad (4)$$

где z, r, φ — координаты цилиндрической системы; $\gamma = \frac{r}{a}$ — относительное удаление оси цилиндра от точки наблюдения.

Анализ выражения (3) показывает, что характер частотной зависимости аномальной функции цилиндрической неоднородности очень близок к аналогичной функции сферической частицы. Поэтому выводы в отношении предпочтительного диапазона рабочих частот для сферической частицы остаются справедливыми и для частицы цилиндрической формы.

Для определения чувствительности приемного устройства системы необходимо определить характер распределения первичного поля в плоскости прямоугольника $AB\Gamma$ (рис. 2), который представляет собой окно для прохождения контролируемого технологического сырья. На рис. 2 показано взаимное расположение генераторной (1), приемной (2) и компенсирующей (3) рамок. Напряженность первичного магнитного поля генераторной рамки в любой внутренней точке можно определить из выражения

$$H_{z0} = \frac{I\omega}{4\pi} \sum_{i=1}^4 \frac{r_i}{S_i}, \quad (5)$$

где I, ω — ток и число витков генераторной рамки; r_i — длины отрезков, соединяющих вершины рамки с точкой наблюдения; S_i — площади прямоугольников, для которых отрезки r_i явля-

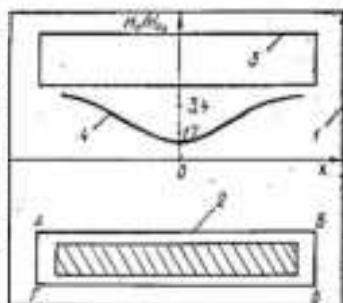


Рис. 2. Распределение первичного поля по профилю окна магнитно — приемника

ются диагоналями. Распределение первичного поля по профилю окна *АВВГ* показано на рис. 2 кривой 4.

Зная распределение первичного поля и значения аномальных функций, представляется возможным определить величину ЭДС, наводимую вторичными полями в приемной рамке. Будем исходить из того, что источником вторичного поля является металлическая частица, размеры которой значительно меньше расстояний до сторон приемной рамки, а вторичное поле внутри рамки не является однородным. ЭДС, наводимая в приемной рамке, определяется скоростью изменения магнитного потока в плоскости рамки

$$E = - \frac{d\psi}{dt} = - \omega \frac{d\Phi}{dt}.$$

Магнитный поток вторичного поля определяется соотношением

$$\Phi = \int_S \mu_0 \omega 2\pi H_z D \frac{d^3}{R^3} dS, \quad (6)$$

где *S* — площадь приемной рамки. ЭДС в приемной рамке для случая, когда металлическая частица сферической формы находится в плоскости витков этой рамки:

$$E = \mu_0 \omega 2\pi H_z D \left(2\pi a^2 - a^3 \sum_{i=1}^4 \frac{\sin \gamma_{i1} + \sin \gamma_{i2}}{R_i} \right), \quad (7)$$

где *R_i*, γ_{in} — координаты точки наблюдения по отношению к сторонам и вершинам рамки.

В том случае, когда металлическое тело сильно вытянутой формы ($l \gg a$) расположено параллельно виткам приемной рамки, то $E \approx 0$. Для надежного обнаружения сильно вытянутых тел независимо от их ориентации по отношению к виткам, необходимо устанавливать две приемные рамки, расположенные под некоторым углом друг к другу.

Приведенный в работе анализ позволяет обоснованно подойти к выбору таких параметров системы очистки технологического сырья, как чувствительность приемника и частоты поля возбуждения. Наличие информации об этих параметрах позволяет определить конструктивные параметры датчика и выходные параметры источника возбуждения.

Поступила в редколлегию 14.11.80.

Н. М. ПОДАРЕВА

ВЫБОР ДОСТАТОЧНОГО СПЕКТРАЛЬНОГО СОСТАВА ВХОДНОГО СИГНАЛА ДЛЯ ИДЕНТИФИКАЦИИ ДИ- НАМИЧЕСКИХ СИСТЕМ

Согласно [1], те входные сигналы, которые позволяют выявить интересующие свойства исследуемого объекта, называются информативными. В данной работе предлагается доказательство достаточности количественного спектрального состава входного сигнала

$$U(t) = \sum_{i=1}^N A_i \sin \omega_i t, \quad (1)$$

где A_i , ω_i — амплитуда и фаза i -й составляющей спектра, для определения $m+n+1$ коэффициентов дифференциального уравнения (ДУ)

$$\theta_{m+1}\dot{\phi} + \theta_{m+2}\dot{\phi}^{(1)} + \dots + \theta_{m+n}\dot{\phi}^{(n-1)} + \dot{\phi}^{(n)} = \theta_0 U + \theta_1 U^{(1)} + \dots + \theta_m U^{(m)} \quad (2)$$

в динамическом и статическом режимах работы объекта. Здесь ϕ — регулируемая координата; U — управляющий сигнал (1).

Запишем ДУ (2) в векторной форме $\dot{\phi}^{(n)} = \theta^T \Phi(t)$ (3), где $\theta = [\theta_0, \theta_1, \dots, \theta_{m+n}]^T$ — вектор идентифицируемых параметров размерности $m+n+1$; $\Phi(t) = [U, U^{(1)}, \dots, U^{(m)}, -\phi, -\dot{\phi}^{(1)}, \dots, -\dot{\phi}^{(n-1)}]^T$ — вектор измерений размерности $m+n+1$.

Информацию об измерениях будем снимать в дискретные моменты времени $j, j+1, \dots, j+v$. По истечении $v+1$ тактов можно записать следующее матричное соотношение:

$$\begin{aligned} & \begin{vmatrix} \dot{\phi}_j^{(n)} \\ \dot{\phi}_{j+1}^{(n)} \\ \dots \\ \dot{\phi}_{j+v}^{(n)} \end{vmatrix} = \\ & = \begin{vmatrix} U_{(j)} & U_{(j)}^{(1)} & \dots & U_{(j)}^{(m)} & -\phi_{(j)} & -\dot{\phi}_{(j)}^{(1)} & \dots & -\dot{\phi}_{(j)}^{(n-1)} \\ U_{(j+1)} & U_{(j+1)}^{(1)} & \dots & U_{(j+1)}^{(m)} & -\phi_{(j+1)} & -\dot{\phi}_{(j+1)}^{(1)} & \dots & -\dot{\phi}_{(j+1)}^{(n-1)} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ U_{(j+v)} & U_{(j+v)}^{(1)} & \dots & U_{(j+v)}^{(m)} & -\phi_{(j+v)} & -\dot{\phi}_{(j+v)}^{(1)} & \dots & -\dot{\phi}_{(j+v)}^{(n-1)} \end{vmatrix} \begin{vmatrix} \theta_0 \\ \theta_1 \\ \dots \\ \theta_{m+n} \end{vmatrix}, \quad (4) \end{aligned}$$

или $\dot{\phi}(j, v) = \Xi(j, v)\theta$ (5), где $\dot{\phi}(j, v)$ — вектор размерности $v+1$; $\Xi(j, v)$ — матрица размерности $(v+1) \times (m+n+1)$.

Известно, что для определения составляющих вектора параметров в размерности $m+n+1$ необходимо, чтобы ранг матрицы $\Xi(j, v)$ был не менее $m+n+1$, для чего число измерений $v+1$ должно быть не менее $m+n+1$ [2]. Это условие является необходимым, но не достаточным для того, чтобы ранг матрицы $\Xi(j, v)$ был равен $m+n+1$. Ранг матрицы определяется числом линейно независимых столбцов и строк в ней. Пусть этому условию удовлетворяет входной сигнал (1) и нас интересует достаточное число N , определяющее ранг матрицы $\Xi(j, m+n)$ как число, равное $m+n+1$.

Утверждение 1. Для определения $m+n+1$ коэффициентов ДУ линейной динамической системы (2) в динамическом режиме функционирования необходимое и достаточное число N гармоник информативного входного сигнала (1) определяется порядком правой части ДУ (2) из соотношения $N > \frac{m+1}{2}$ (6).

Доказательство. Для доказательства получим аналитические выражения для составляющих матрицы измерений $\Xi(j, m+n)$, используя аппарат преобразований Лапласа. Изображение по Лапласу входного сигнала (1) согласно [2] имеет вид

$$U(p) = \sum_{i=1}^N A_i \frac{\omega_i}{p^2 + \omega_i^2} = \frac{M(p)}{R(p)}. \quad (7)$$

Передаточная функция объекта (2) может быть представлена так:

$$W(p) = \frac{\psi(p)}{U(p)} = \frac{\psi_0 + \psi_1 p + \dots + \psi_m p^m}{1 + \psi_{m+1} p + \dots + \psi_{m+n} p^n} = \frac{B(p)}{A(p)}. \quad (8)$$

С учетом выражений (7) и (8) изображение выходного сигнала можно найти как

$$\psi(p) = \frac{B(p)}{A(p)} U(p) = \frac{B(p) M(p)}{A(p) R(p)}. \quad (9)$$

Возьмем наиболее общий случай, когда полюса (9): $Q_1, Q_2, \dots$ являются простыми и некрратными. Тогда при разложении (9) на сумму простейших дробей [2] можно получить

$$\psi(p) = \sum_{i=1}^n g_i \frac{1}{p - Q_i} + \sum_{j=1}^N c_j \frac{p}{p^2 + \omega_j^2} + \sum_{k=1}^N h_k \frac{1}{p^2 + \omega_k^2}, \quad (10)$$

где g_i, c_j, h_k — некоторые постоянные величины. Тогда выражение для представления оригинала выходного сигнала (10) можно записать согласно [2] как

$$\begin{aligned} y(t) &= L^{-1} \{ \psi(p) \} = \sum_{i=1}^n g_i e^{Q_i t} + \sum_{j=1}^N c_j \sin \omega_j t + \sum_{k=1}^N h_k \cos \omega_k t = \\ &= G_0^T Z(t) + C_0^T R(t) + H_0^T Y(t). \end{aligned} \quad (11)$$

В полученном выражении

$$\begin{aligned} G_0 &= [g_{1,0}, g_{2,0}, \dots, g_{N,0}]^T, \quad Z(t) = [e^{Q_1 t}, e^{Q_2 t}, \dots, e^{Q_N t}]^T = \\ &= [z_1(t), z_2(t), \dots, z_N(t)]^T, \quad C_0 = [c_{1,0}, c_{2,0}, \dots, c_{N,0}]^T, \\ R(t) &= [\sin \omega_1 t, \sin \omega_2 t, \dots, \sin \omega_N t]^T = [r_1(t), r_2(t), \dots, r_N(t)]^T, \\ H_0 &= [h_{1,0}, h_{2,0}, \dots, h_{N,0}]^T, \quad Y(t) = [\cos \omega_1 t, \cos \omega_2 t, \dots, \\ &\quad \cos \omega_N t]^T = [y_1(t), y_2(t), \dots, y_N(t)]^T. \end{aligned} \quad (12)$$

— векторы, размерность которых очевидна. Второй индекс появился искусственно как свидетельство о номере производной, что необходимо для общности изложения в дальнейшем. На основании (11) легко получить производные выходного сигнала:

$$\begin{aligned} \varphi^{(1)}(t) &= G_1^T Z(t) + C_1^T R(t) + H_1^T Y(t), \\ \varphi^{(2)}(t) &= G_2^T Z(t) + C_2^T R(t) + H_2^T Y(t), \\ \varphi^{(n-1)}(t) &= G_{n-1}^T Z(t) + C_{n-1}^T R(t) + H_{n-1}^T Y(t). \end{aligned} \quad (13)$$

Векторы G_i , C_i , H_i здесь определяются аналогично (12).

По известной аналитической зависимости входного сигнала от времени (1) можно получить выражения для его производных:

$$\begin{aligned} U(t) &= \sum_{i=1}^N A_{i,0} e^{r_i(t)} = A_0^T R(t), \\ U^{(1)}(t) &= \sum_{i=1}^N A_{i,1} y_i(t) = A_1^T Y(t), \\ &\vdots \\ U^{(m)}(t) &= \sum_{i=1}^N A_{i,m} r_i(t) = A_m^T R(t), \text{ если } m \text{ четное,} \\ U^{(m)}(t) &= \sum_{i=1}^N A_{i,m} y_i(t) = A_m^T Y(t), \text{ если } m \text{ нечетное,} \end{aligned} \quad (14)$$

Здесь $A_j = [A_{1,j}, A_{2,j}, \dots, A_{N,j}]^T$, ($j = 0, m$) — векторы амплитуд производных размерностей N .

Представим матрицу $\Xi(j, m+n)$ в виде произведения двух матриц: $\Xi(j, m+n) = \Gamma(j, m+n) \cdot \Delta$ (15), где

$$\Gamma(j, m+n) = \begin{bmatrix} R_{(j)}^T & Y_{(j)}^T & Z_{(j)}^T \\ R_{(j+1)}^T & Y_{(j+1)}^T & Z_{(j+1)}^T \\ \vdots & \vdots & \vdots \\ R_{(j+m+n)}^T & Y_{(j+m+n)}^T & Z_{(j+m+n)}^T \end{bmatrix}. \quad (16)$$

— клеточная матрица размерности $(m+n+1) \times (2N+n)$;

$$\Delta = \begin{vmatrix} A_0 & 0 & A_2 & \dots & A_m & C_0 & C_1 & \dots & C_{n-1} \\ 0 & A_1 & 0 & \dots & 0 & H_0 & H_1 & \dots & H_{n-1} \\ 0 & 0 & 0 & \dots & 0 & G_1 & G_2 & \dots & G_n \end{vmatrix}. \quad (17)$$

— клеточная матрица размерности $(2N+n) \times (2m+n+1)$.

Ранг матрицы (15) должен быть равен числу определяемых параметров $m+n+1$. С другой стороны, ранг произведения двух матриц (15) по условию Сильвестра [2] не может быть больше минимального ранга матриц-составляющих, следовательно, ранги матриц Γ и Δ должны удовлетворять условию

$$r_1 \geq m+n+1, \quad r_2 \geq m+n+1. \quad (18)$$

Ранги матриц-составляющих определяются по числу линейно независимых строк и столбцов в них:

$$r_1 = \min(m+n+1, 2N+n), \quad r_2 = \min(2N+n, 2m+n+1). \quad (19)$$

Очевидно, что r_2 может быть больше r_1 при $m > 0$, поэтому ранг матрицы (15) определяется рангом матрицы Γ . Но поскольку число ее строк равно $m+n+1$, то при любом увеличении числа ее столбцов, зависящего от N , $2N+n \geq m+n+1$ (20), ранг матрицы (16) не будет больше, чем $m+n+1$. Из условия (20) следует, что $2N \geq m+1$, откуда $N \geq (m+1)/2$, что и требовалось доказать.

Утверждение 2. Для определения $m+n+1$ коэффициентов ДУ устойчивой динамической системы (2) при ее функционировании в установившемся режиме число составляющих спектра входного сигнала (1) определяется порядками правой и левой частей ДУ из соотношения $N \geq \frac{m+n+1}{2}$.

Доказательство. Рассмотрим матрицу (16). В установившемся режиме векторы $Z(j) \approx Z(j+1) \approx \dots \approx Z(j+m+n) \approx 0$; тогда ранг матрицы (16) определится как $r_1 = \min(m+n+1, 2N)$. Поскольку должно выполняться условие $r_1 \geq m+n+1$, то $2N \geq m+n+1$, откуда $N \geq (m+n+1)/2$, что и требовалось доказать.

Доказанные утверждения определяют требования к достаточному составу спектра входного сигнала. В работе [3] приведены некоторые номограммы и рекомендации по выбору значений параметров гармонических составляющих в зависимости от собственной частоты объекта и частоты опроса, предложенные на основании эксперимента.

Список литературы: 1. Костюк В. И. Адаптивные системы идентификации. — К.: Техника, 1975. — 288 с. 2. Корн Г., Корн Т. Справочник по математике. — М.: Наука, 1975. — 685 с. 3. Соколов Ю. Н., Подарева Н. М., Успенко В. И. Информативность тестовых сигналов в задаче идентификации параметров объекта с настраиваемой моделью. — В кн.: Системы управления летательных аппаратов, вып. 3. Х.: ХАН. 1977, с. 31–38.

Поступила в редколлегию 29.12.80.

ОПРЕДЕЛЕНИЕ КОЛИЧЕСТВА ОЦЕНИВАЕМЫХ ПАРАМЕТРОВ В ЗАДАЧАХ ИДЕНТИФИКАЦИИ С КОРРЕЛИРОВАННЫМИ ПОМЕХАМИ

Предположим, что рассматривается динамический процесс с участками стационарности. Относительно этого процесса известно, что при числе измерений n и интервале дискретизации ΔT таком, что $\frac{T}{\Delta T} > n$, измерения $\{y_i\}_{i=1}^n$ этого процесса коррелированы. Этот процесс будем оценивать на стационарных участках с помощью регрессионной модели, линейной по параметрам, причем количество параметров m достаточно велико, хотя и $m \leq n$.

Подобная ситуация может встретиться на установках в производстве масел нефтеперерабатывающего завода, когда при числе входных переменных, равном 16, в модель включаются линейные члены и квадраты (число оцениваемых параметров равно 32), а объем информации равен 74.

Приведем математическую постановку задачи.

Пусть оценивается модель: $y_i = \theta^* f(x_i) + \xi_i$, $i = 1, 2, \dots, n$, (1), $\theta \in R^m$ — оцениваемые параметры, $x \in X \subset R^k$, $f^*(x) = (f_1(x), \dots, f_m(x))$ непрерывная в области X вектор-функция; $\{\xi_i\}_{i=1}^n$ — случайные нормальные составляющие с нулевым средним и корреляционной матрицей: $R_n = \{r_{ij}\}_{i,j=1}^n$.

Матрица R_n заранее неизвестна. Оценку матрицы R_n можно осуществить с помощью оценки корреляционной функции стационарного процесса по формуле [1]

$$\hat{r}(\tau) = \frac{1}{n-\tau} \sum_{i=1}^{n-\tau} (y(i+\tau) - \bar{y})(y(i) - \bar{y}). \quad (2)$$

Эта оценка, главным образом, зависит от количества измерений n , и, кроме того, ошибка в определении ее тем больше, чем больше τ . Оптимальная линейная оценка θ по n измерениям имеет вид $\hat{\theta}_n = (F_n^* R_n^{-1} F_n)^{-1} F_n^* R_n^{-1} Y_n$ (3), где

$$F_n^* = [f(x_1), f(x_2), \dots, f(x_n)], \quad Y_n = (y_1, y_2, \dots, y_n).$$

Ясно, что при большом m необходимость обращения матриц в формуле (3) приводит к существенным ошибкам, а, кроме того, при плохой оценке матрица R_n может быть вырождена.

Для преодоления указанных трудностей предлагается перейти от коррелированных помех к некоррелированным с помощью оценки (2) по формулам [2]:

$$Y = AZ, \quad A = \begin{pmatrix} 1 & 0 & \dots & 0 \\ a_{21} & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & 1 \end{pmatrix}, \quad a_{s1} = \frac{r_{s1}}{Dz_1},$$

$$Dz_i = r_{ii} - \sum_{\lambda=1}^{i-1} |a_{i\lambda}|^2 Dz_{\lambda}, \quad (4)$$

$$a_{\mu\lambda} = \frac{1}{Dz_{\lambda}} \left(r_{\mu\lambda} - \sum_{i=1}^{\lambda-1} a_{\mu i} a_{\lambda i} Dz_i \right), \quad \mu < \nu,$$

где z_i — некоррелированные измерения, Dz_i — дисперсия z_i . Это же преобразование можно провести рекуррентно с помощью формул [3]. При этом может оказаться, что для некоторого $\mu < n$ $Dz_{\mu} = 0$. Это возможно за счет недостаточно точной оценки корреляционной матрицы R_n . Если $\mu < m$ это означает, что на данном этапе мы можем оценить только μ из m параметров. Поэтому при построении модели необходимо вначале провести исследование возможного числа оцениваемых параметров, а затем уже оценивать саму модель.

Ясно, что при оценивании модели следует учесть, что преобразование (4) нужно применить и к функциям $f_i(x)$.

Для выбора μ из m оцениваемых параметров можно воспользоваться методом структурной идентификации, либо методом ранговой корреляции для переменных Z, V , где $AV_i = f_i(x_i)$.

В качестве примера рассматривалась задача построения регрессионной линейной модели по 14 переменным. Для этого использовалась выборка выходной переменной промышленного объекта, по которой оценивалась корреляционная матрица и далее производился переход к некоррелированным переменным по формулам (4). Результаты расчетов на ЭВМ приведены в таблице.

N	1	2	3	4	5	6	7	8	9	10	11
Z	-2,25	0,49	-0,30	-0,81	1,25	-1,51	-2,19	0,22	0,19	0,82	1,03
DZ	0,45	0,12	0,09	0,13	0,31	0,27	0,16	0,08	0,07	0,01	-0,50

Из приведенных результатов вытекает, что $Dz_{11} < 0$, поэтому здесь мы можем оценить не более 10 параметров и, следовательно, возникает необходимость выбора наиболее существенных. При исследовании выяснилось, что из 14 переменных в линейном случае при идентификации линейной модели по 20 измерениям можно оценить только 10 параметров из 14, и эти параметры выбираются оптимальным образом одним из методов структурной идентификации [4].

Список литературы: 1. Бендат Дж., Пирсол А. Измерение и анализ случайных процессов. — М.: Мир, 1974. — 646 с. 2. Пугачев В. С. Теория случайных функций. — М.: Физматгиз, 1960. — 883 с. 3. Аюбьян Э. Д. Рекуррентный метод наименьших квадратов при коррелированных помехах. — Автоматика и телемеханика, 1975, № 5, с. 67—75. 4. Забара Н. М., Коц Н. П., Мелешко В. И., Назырова В. П. Об одном алгоритме оптимизации структуры модели в задачах статистической идентификации. Тез. докл. (Совещание по применению случайного поиска для решения прикладных задач). Кемерово, 1979, с. 37—38.

Поступила в редакцию 20.06.80.

ИССЛЕДОВАНИЕ РЕКУРРЕНТНЫХ СТОХАСТИЧЕСКИХ АЛГОРИТМОВ ИДЕНТИФИКАЦИИ С ПОСТОЯННЫМ ШАГОМ И НЕГЛАДКИМИ ФУНКЦИЯМИ РЕГРЕССИИ

В статье рассматриваются предельные теоремы для алгоритмов идентификации, основанных на идее метода стохастической аппроксимации, суть которых состоит в следующем: найти x^* — единственное решение уравнения $R(x) = a$ по наблюдениям случайного процесса

$$dy(x, t) = \rho [R(x) + a(\rho) W(t, \rho, x)] dt + W(\rho) D(t, \rho, x) dw(t), t > 0,$$

где ρ — постоянный шаговый множитель, $\rho > 0$; $R(x)$ — N -мерная вектор-функция регрессии; $R(x)$, $W(t, \rho, x)$, $D(t, \rho, x)$ — борелевские функции по $x \in R^N$ — N -мерному евклидовому пространству; $w(t)$ — стандартный N -мерный винеровский процесс, в работе рассматриваются вектор-столбцы. Для решения этой задачи рассмотрим уравнение

$$dx_r(t) = dy(x_r(t), t) - \rho x dt, t > 0 \quad (1), \quad x_r(0) = \zeta \quad (2),$$

где ζ — случайный вектор, не зависящий от $w(t)$, $M \|\zeta\|^2 < \infty$,

$$\psi(\rho) = \rho^*, \quad x > -1/2, \quad \alpha(\rho) = \rho^*, \quad \alpha \geq 1/2 + x. \quad (3)$$

Как следует из [1], в том случае, когда уравнение (1), (2) описывает процессы функционирования некоторых самонастраивающихся систем, величина шага ρ имеет физический смысл коэффициента передачи. Там же отмечалось, что убывание коэффициента передачи при $t \rightarrow \infty$, что имеет место в случае алгоритмов с переменным шагом [2], нежелательно, поскольку «постоянная обратная связь обеспечивает более быстродействующий процесс, чем обратная связь, исчезающая с течением времени» [1, с. 76]. Процедура вида (1), (2) соответствует автоматической системе с постоянным коэффициентом передачи.

Предполагается, что выполняются все условия [3], обеспечивающие существование, единственность и непрерывность решения уравнения (1), (2). А именно,

$$\sup_{t \geq 0} \rho \{ \|R(x) - R(y)\| + \alpha(\rho) \|W(t, \rho, x) - W(t, \rho, y)\| + \\ + \psi(\rho) \|D(t, \rho, x) - D(t, \rho, y)\| \} \leq k \|x - y\|, \quad (4)$$

где

$$\|x\| = \langle x, x \rangle^{1/2}, \quad \langle x, y \rangle = \sum_{i=1}^N [x]_i [y]_i, \quad [x]_i =$$

i -я координата вектора x , норма матрицы определяется так: $\|B\| = \max_{\|x\|=1} \langle BB^*x, x \rangle$, штрихом обозначается транспонирование.

Приведем дальнейшие предположения о вектор-функции $R(x)$. Не ограничивая общности, будем считать $x^* = 0$, $\alpha = 0$.

Пусть пространство R^N разбито гиперплоскостями, проходящими через точку 0 на K многогранных конусов Γ_i и

$$\bigcup_{i=1}^K \Gamma_i = R^N, \quad \Gamma_i \cap \Gamma_j = \partial\Gamma_i \cap \partial\Gamma_j \text{ при } i \neq j.$$

Предположим, что $R(x) = A(x)x + \delta(x)$ (5), где $\|\delta(x)\| = o(\|x\|)$ при $x \rightarrow 0$, $A(x)$ — матричнозначная функция следующего вида: при $x \in \text{Int } \Gamma_i$ $A(x) = A_i$, $i = \overline{1, K}$, A_i — устойчивые матрицы, $\min_{i=\overline{1, K}} \text{Re } \lambda_j(A_i) = \omega > 0$, $\lambda_j(A_i)$, $j = \overline{1, N}$ — собственные числа матрицы A_i .

$$A(0) = \frac{1}{K} \sum_{i=1}^K A_i, \quad A(x) = \frac{1}{m} \sum_{i=1}^m A_{i_l}, \quad 2 \leq m \leq K, \quad i_l \in \{1, \dots, K\},$$

при $x \in \bigcap_{i=1}^K \partial\Gamma_i \setminus \{0\}$; $A_i x = A_i x$ при $x \in \partial\Gamma_i \cap \partial\Gamma_j$, $\partial\Gamma_i$ — граница множества Γ_i , $\text{Int } \Gamma_i$ — внутренность множества Γ_i .

Заметим, что в силу сделанных предположений функция удовлетворяет условию Липшица, т. е. $\forall x, y \in R^N$

$$\|A(x)x - A(y)y\| \leq k\|x - y\|. \quad (6)$$

Здесь и дальше k — произвольные постоянные, не обязательно одинаковые, величина которых не существенна.

Замечание 1. Примером одномерной функции, удовлетворяющей перечисленным условиям является следующая:

$$f(x) = \begin{cases} \alpha_1 x, & \alpha_1 < 0, \alpha_1 \neq \alpha_2 \text{ при } x < 0, \\ \frac{1}{2}(\alpha_1 + \alpha_2) & \text{при } x = 0, \\ \alpha_2 x, & \alpha_2 < 0 \end{cases} \quad \text{при } x > 0.$$

Приведем теорему, обеспечивающую среднеквадратическую устойчивость процедуры (1), (2) по величине шага p , а следовательно, в силу неравенства Маркова [4], устойчивость по вероятности. При этом под устойчивостью по величине шага p здесь понимается свойство $x_2(t)$ сближаться с решением уравнения $R(x) = 0$ при уменьшении величины шага p для соответственно больших t .

Теорема 1. [5] Пусть выполняются следующие условия:

$$1) \sup_{t \geq 0} [\|R(x)\| + \|W(t, p, x)\| + \|D(t, p, x)\|] \leq k(1 + \|x\|);$$

2) существует симметричная положительно определенная матрица C такая, что

$$\langle CR(x), x \rangle \leq -\lambda \langle Cx, x \rangle, \quad \lambda > 0.$$

Тогда существует $\rho_0 > 0$ такое, что при $\rho < \rho_0$

$$\limsup_{t \rightarrow \infty} M \|x_\rho(t)\|^2 < k\rho.$$

Следствие 1. [5] В силу неравенства Маркова, при выполнении условий теоремы 1, получим для всякого $\epsilon > 0$

$$\lim_{\rho \rightarrow 0} \lim_{t \rightarrow \infty} P \{ \|x_\rho(t)\| > \epsilon \} = 0,$$

т. е. процедура (1), (2) устойчива по вероятности.

Замечание 2. В математическом программировании источником условий типа 2 теоремы 1 является следующий факт. Если функция, определенная на выпуклом замкнутом множестве в R^N , сильно выпукла, то ее антиградиент удовлетворяет условию 2 теоремы 1. Причем в этом случае $C = E$ — единичная матрица в R^N .

Замечание 3. Если матрица A устойчива и нормальна, то она отрицательно определена, т. е. $\max_{\|x\|=1} \langle Ax, x \rangle < 0$. Поэтому, если $A_i, i = \overline{1, K}$ нормальны, то в качестве C можно взять единичную матрицу E в R^N .

Рассмотрим случайные процессы

$$x_{\rho t}(v) = \rho^{-1/2} x_\rho(t + v/\rho), \quad v \in [0, T], \quad T \in (0, \infty). \quad (7)$$

Вопрос о существовании предельного распределения у $x_{\rho t}(\cdot)$ при повторном стремлении $\rho \rightarrow 0, t \rightarrow \infty$ в случае негладкой функции $R(x)$ вида (5) является нерешенным, как в случае переменного шага, так и постоянного. Здесь покажем, что предельное распределение существует, однако в отличие от процессов, построенных по процедурам с функцией регрессии, дифференцируемой в точке 0, не является, вообще говоря, гауссовским.

Теорема 2. Пусть $R(x)$ есть функция вида (5), выполняются условия теоремы 1 и следующие:

$$3) \lim_{\max(t^{-1}, \rho, \|x\|) \rightarrow 0} D(t, \rho, x) = D, \quad \det D \neq 0;$$

$$4) \alpha > 1/2 + \chi;$$

или

$$4') \alpha = 1/2 + \chi, \quad \lim_{\max(t^{-1}, \rho, \|x\|) \rightarrow 0} W(t, \rho, x) = W.$$

Тогда распределение $\varphi(x_{\rho t}(\cdot))$ при повторном стремлении $\rho \rightarrow 0, t \rightarrow \infty$ сходится к распределению $\varphi(z(\cdot))$ для любого измеримого и непрерывного на $C^N[0, T]$ функционала $\varphi(\cdot)$ со значениями в R^q . Здесь $C^N[0, T]$ — пространство непрерывных N -мерных вектор-функций с равномерной метрикой, заданных на $[0, T]$, а $z(\cdot)$ — стационарный марковский процесс вида

$$dz(t) = [A(z(t))z(t) + U]dt + Ddw(t), \quad z(0) = z_0,$$

где z_0 и $w(t)$, $t > 0$ независимы, U — неслучайный вектор в R^N , $U = 0$ при выполнении условия 4, а при выполнении условия 4' $U = W$.

Если же $A_i A_j = A_j A_i$, $i \neq j$, $i, j = \overline{1, K}$, то $z(\cdot)$ стохастически эквивалентен случайному процессу $Z(\cdot)$ вида

$$Z(t) = \exp \left\{ \int_0^t A(Z(s)) ds \right\} \left\{ Z(0) + \int_0^t \exp \left\{ - \int_0^s A(x(\tau)) d\tau \right\} \times \right. \\ \left. \times [U ds + D dw(s)], \right.$$

где Z_0 не зависит от приращений $w(t)$, $t > 0$ и распределен так же, как предел при $t \rightarrow \infty$ в смысле слабой сходимости случайных векторов

$$\int_0^t \exp \left\{ \int_0^s A(Z(\tau)) d\tau \right\} [U ds + D dw(s)].$$

Доказательство. Рассмотрим случай, когда выполнено условие 4'. Случай, когда выполнено условие 4 рассматривается аналогично. Непосредственной подстановкой в уравнение (1) нетрудно убедиться в том, что случайные процессы $x_{pt}(\cdot)$ удовлетворяют уравнению

$$dx_{pt}(v) = b_{pt}(v, x_{pt}(v)) dv + D_{pt}(v, x_{pt}(v)) dw_{pt}(v), \quad (8)$$

где $w_{pt}(v) = p^{1/2} [\omega(t + v/p) - \omega(t)]$ — стандартный N -мерный винеровский процесс,

$$b_{pt}(v, x) = p^{-(1/2+\epsilon)} [R(p^{1/2+\epsilon}x) + p^2 W(t + v/p, p, p^{1/2+\epsilon}x)],$$

$$D_{pt}(v, x) = D(t + v/p, p, p^{1/2+\epsilon}x).$$

Так как выполнены условия теоремы 1, то

$$\overline{\lim}_{p \rightarrow 0} \overline{\lim}_{t \rightarrow \infty} M \|x_{pt}(0)\|^2 = \overline{\lim}_{p \rightarrow 0} \overline{\lim}_{t \rightarrow \infty} M \|p^{-(1/2+\epsilon)}x(t)\|^2 \leq k. \quad (9)$$

Из соотношений (8) — (9), в силу результатов книги [6], вытекает, что меры, отвечающие $x_{pt}(\cdot)$ слабо компакты в $C^N[0, T]$ при повторном стремлении $p \rightarrow 0$, $t \rightarrow \infty$. Таким образом, осталось доказать сходимость частных распределений $x_{pt}(\cdot)$.

Применим процедуру усечения к функциям $b_{pt}(v, x)$, $D_{pt}(v, x)$ такую же, как в работе [7]. В результате получим функции $b_{pt}(v, x, \lambda)$, $D_{pt}(v, x, \lambda)$, где $\lambda > 0$ — произвольное, как угодно большое число. Так как функции $b_{pt}(v, x, \lambda)$, $D_{pt}(v, x, \lambda)$, в силу (4) удовлетворяют условию Липшица, то решение уравнения

$$dy_{pt}(v) = b_{pt}(v, y_{pt}(v), \lambda) dv + D_{pt}(v, y_{pt}(v), \lambda) \times \\ \times dw_{pt}(v), \quad y_{pt}(0) = x_{pt}(0)$$

существует, непрерывно с вероятностью 1 и является единственным. Кроме того, при $t > t(p)$

$$P\{y_{pt}(v) \neq x_{pt}(v), v \in [0, T]\} < \frac{k}{\lambda^2}.$$

Отсюда, вследствие того, что λ может быть произвольно большим, вытекает, что при повторном стремлении $\rho \rightarrow 0$, $t \rightarrow \infty$ меры, соответствующие в $C^N[0, T]$ случайным процессам $x_{\rho t}(\cdot)$ и $y_{\rho t}(\cdot)$, ведут себя одинаково. Поэтому покажем, что случайные процессы $y_{\rho t}(y)$ слабо сходятся при повторном стремлении $\rho \rightarrow 0$, $t \rightarrow \infty$ к случайному процессу $z(\cdot)$.

Из способа построения $b_{\rho t}(v, x, \lambda)$ и $D_{\rho t}(v, x, \lambda)$ вытекает, что

$$\begin{aligned} \|b_{\rho t}(v, x, \lambda) - [A(x)x + W]\| &\leq \|x\| \sup_{\|x\| \leq \rho^{1/2}t + v_1} \|\delta(x)\| \times \\ &\times \|x\|^{-1} + \sup_{\substack{\|x\| \leq \rho^{1/2}t + v_1 \\ s \geq t}} \|W(t, \rho, x) - W\| = \|x\| \delta(\rho) + \delta(\rho, t), \\ \|D_{\rho t}(v, x, \lambda) - D\| &\leq \sup_{\substack{\|x\| \leq \rho^{1/2}t + v_1 \\ s \geq t}} \|D(s, \rho, x) - D\| = d(\rho, t). \end{aligned}$$

В силу условий 3 и 4' теоремы, при повторном стремлении $\rho \rightarrow 0$, $t \rightarrow \infty$

$$\delta(\rho) + \delta(\rho, t) + d(\rho, t) \rightarrow 0. \quad (10)$$

Из (9) вытекает ограниченность в среднем квадратичном при $\rho \rightarrow 0$, $t \rightarrow \infty$, а, следовательно, в силу неравенства Маркова, стохастическая ограниченность случайных векторов $x_{\rho t}(0)$. Поэтому у $x_{\rho t}(0)$ при $\rho \rightarrow 0$, $t \rightarrow \infty$ существует некоторое число предельных точек в смысле слабой сходимости. Обозначим их через $x(0, \alpha)$, $\alpha \in A$, а через P_α и T_α — множества положительных чисел таких, что при повторном стремлении ρ по P_α к 0, а t по T_α к ∞ в пределе получается $x(0, \alpha)$. Здесь A — некоторое множество индексов. Тогда для $\rho \in P_\alpha$ и $t \in T_\alpha$ при $\rho \rightarrow 0$, $t \rightarrow \infty$

$$x_{\rho t}(0) \xrightarrow{cl} x(0, \alpha). \quad (11)$$

В силу теоремы о сходимости моментов и оценки (9)

$$M \|x(0, \alpha)\|^2 \leq k^{N/2}, \quad \forall \alpha \in (0, 2). \quad (12)$$

Соотношения (9), (10) — (12) показывают, что применима теорема 14 [6, с. 277], из которой следует сходимость в $C^N[0, T]$ мер, отвечающих случайным процессам $x_{\rho t}(\cdot)$, $\rho \in P_\alpha$, $t \in T_\alpha$ при $\rho \rightarrow 0$, $t \rightarrow \infty$ к мере, отвечающей процессу $y^\alpha(\cdot)$ вида

$$dy^\alpha(t) = [A(y^\alpha(t))y^\alpha(t) + W]dt + Ddw(t), \quad (13)$$

$$y^\alpha(0) = x(0, \alpha), \quad (14)$$

где $x(0, \alpha)$ не зависит от приращений $w(t)$, $t > 0$.

Используя результаты [8, 9], можно доказать существование единственного стационарного распределения $\mu(\cdot)$ случайного процесса (13), (14), такого, что при фиксированных $\rho > 0$, $\delta \in (0, 2)$

и $u \in R^N$ можно выбрать $\tilde{t} = \tilde{t}(\delta, u, k^{N/2})$, что при $t > \tilde{t}$ для всех $\alpha \in A$

$$\|M \exp\{t < y^\alpha(t), u\} - H(u)\| < \delta, \quad (15)$$

где $l = \sqrt{-1}$, $H(\cdot)$ — характеристическая функция стационарного распределения $\mu(\cdot)$. Заметим, что при $t - h/\rho > t(\rho)$ выполнено неравенство, $\rho > 0$, $h > 0$, $y_{\rho t}(0) = y_{\rho t - h/\rho}(h)$. Отсюда для

$$t \geq t(\rho) + h/\rho \quad y_{\rho t}^*(0) = y_{\rho t - h/\rho}^*(h), \quad (16)$$

где $y_{\rho t}^*(v) = y_{\rho t}(v)$ при $\rho \in P_*$, $t \in T_*$.

Так как $\{y_{\rho t - h/\rho}^*(0)\}$ составляют подмножество случайных векторов $\{y_{\rho t}(0)\}$, то в силу стохастической ограниченности при повторном стремлении $\rho \rightarrow 0$, $t \rightarrow \infty$ ($\rho \in P_*$, $t \in T_*$) существует хотя бы одна предельная точка $y(0, \beta)$, $\beta \in A$, такая, что

$$y_{\rho t - h/\rho}^*(0) \xrightarrow[\rho \rightarrow 0, t \rightarrow \infty]{c.a.} y(0, \beta),$$

где $\rho \in \tilde{P}_*$, $t \in \tilde{T}_*$, $\tilde{P}_*$, $\tilde{T}_*$ — подмножества P_* и T_* , приводящие к $y(0, \beta)$. С другой стороны, $y_{\rho t}^*(0) \xrightarrow{c.a.} y(0, \alpha)$ при $\rho \rightarrow 0$, $t \rightarrow \infty$. Переходя в (16) к пределу при $\rho \rightarrow 0$, $t \rightarrow \infty$, получим, что $y(0, \alpha)$ имеет такое же распределение, как $y^*(t)$. Следовательно, с учетом (15)

$$|M \exp(i < y(0, \alpha), u > - H(u))| < \varepsilon.$$

Это означает, что для всех $\alpha \in A$ $y(0, \alpha)$ имеет такое же распределение, как $\mu(\cdot)$. Поэтому, уравнение (13), (14) не зависит от α . Следовательно, меры, соответствующие в $C^N[0, T]$ случайным процессам $y_{\rho t}(\cdot)$, а значит $x_{\rho t}(\cdot)$ слабо сходятся к мере, соответствующей случайному процессу $z(\cdot)$. Этот результат эквивалентен утверждению теоремы [6]. Вторая часть теоремы доказывается аналогично соответствующему утверждению работы [10]. Теорема доказана.

Можно найти предельные распределения некоторых функционалов от $x_{\rho t}(\cdot)$, используя результаты работы [11].

Замечание 4. Отметим, что несколько более сложный, чем обычно в работах по стохастической аппроксимации вид случайного направления, фигурирующего в соотношении (1) вызван желанием охватить возможно более широкий класс алгоритмов. Например, при $W(\cdot, \cdot, \cdot) = 0$, $x = 0$ получаются процедуры типа Роббинса — Монро, при $x < 0$ — процедуры типа Кифера — Вольфовица, а при $x > 0$ некоторые алгоритмы случайного поиска. Отметим, что в случае процедуры Кифера — Вольфовица приведенная теорема позволяет доказывать слабую сходимость мер для один раз дифференцируемых функций цели, а не дважды дифференцируемых, как это делается обычно.

Рассмотрим более подробно одномерные процедуры. Пусть $A(x)$ такая же, как $f(x)$, описанная в замечании 1, $D = \beta > 0$. Следуя книге [3], положим

$$r(x) = \int_0^x \exp\{f(z)z^2\} \beta^{-2} dz.$$

Нетрудно проверить, что $r(\cdot)$ монотонно возрастает, поэтому существует $g(\cdot)$ обратная к $r(\cdot)$ функция. Случайный процесс $\eta(t) = r(z(t))$ удовлетворяет соотношению

$$d\eta(t) = \sigma(\eta(t)) d\omega(t), \quad \eta(0) = r(z(0)),$$

где $\sigma(x) = \beta r'(g(x))$, $r'(\cdot)$ — производная $r(\cdot)$. Несложно проверить, что

$$0 < D = \int_{-\infty}^{\infty} \sigma^{-2}(x) dx < \infty.$$

Поэтому в силу леммы 3 [3, с. 138]

$$\int_{-\infty}^{\infty} \frac{F_{x,t}(y)}{\sigma^2(x)} dx = \int_{-\infty}^y \frac{dx}{\sigma^2(x)},$$

где $F_{x,t}(y) = P\{\eta(h+t) < y | \eta(h) = x\}$, $h > 0$. Пусть $F(y) = \mu(-\infty, g(y))$, тогда можно показать, что

$$F(y) = D^{-1} \int_{-\infty}^y \frac{dx}{\sigma^2(x)}. \quad (17)$$

Полагая $T(y) = \mu(-\infty, y) = F(r(y))$ и делая замену $z = g(x)$, получим

$$T(y) = D^{-1} \beta^{-1} \int_{-\infty}^y [r'(z)]^{-1} dz.$$

Отсюда следует, что распределение $T(y)$ имеет плотность следующего вида:

$$t(y) = \beta^{-1} \left(\sqrt{\frac{\pi}{\alpha_1}} + \sqrt{\frac{\pi}{\alpha_2}} \right)^{-1} \exp\{-\beta^{-2} f(y) y^2\}, \quad (18)$$

При этом можно подсчитать, учитывая (18), что

$$Mz_0 = \frac{\beta \sqrt{\pi}}{4} \left(\frac{1}{\sqrt{\alpha_1}} - \frac{1}{\sqrt{\alpha_2}} \right).$$

Это в силу теоремы о сходимости моментов [4] означает, что в случае гладких функций регрессии

$$(\alpha_1 = \alpha_2) \lim_{\beta \rightarrow 0} \lim_{t \rightarrow \infty} Mx_{\beta t}(0) = 0,$$

а в случае негладких функций регрессии

$$(\alpha_1 \neq \alpha_2) \lim_{\beta \rightarrow 0} \lim_{t \rightarrow \infty} Mx_{\beta t}(0) = Mz_0 \neq 0.$$

Таким образом, смещение последовательных приближений в негладком случае асимптотически больше чем в гладком.

Список литературы: 1. Цыпкин Я. З. Адаптация и обучение в автоматических системах. — М.: Наука, 1968. — 400 с. Невельсон М. Б., Хасьямский Р. З. Стохастическая аппроксимация и рекуррентное оценивание. — М.: Наука,

1972.—304 с. 3. Гихман И. И., Скороход А. В. Стохастические дифференциальные уравнения.—К.: Наук. думка, 1968.—354 с. 4. Лом М. Теория вероятностей.—М.: Изд-во иностр. лит., 1962.—719 с. 5. Каниовская И. Ю. Об одной процедуре типа стохастической аппроксимации.—Стохастические модели оптимизации.—К., 1979, с. 37—40. (Препринт 79—13. Ин-т кибернетики АН УССР). 6. Гихман И. И., Скороход А. В. Теория случайных процессов.—М.: Наука, 1975, Т. 3.—496 с. 7. Каниовский Ю. М. О предельном поведении случайных процессов, порожденных процедурой Роббинса—Монро.—Теория вероятностей и мат. статистика, 1978, вып. 19, с. 81—87. 8. Хасминский Р. З. Устойчивость систем дифференциальных уравнений при случайных возмущениях их параметров.—М.: Наука, 1969.—366 с. 9. Каниовская И. Ю. Предельные теоремы для рекуррентных алгоритмов адаптации с негладкими функциями регрессии.—Вероятностные методы в кибернетике.—К., 1979, с. 57—65. (Препринт 79—69. Ин-т кибернетики АН УССР). 10. Каниовская И. Ю. Сходимость случайных процессов, порожденных одной процедурой типа стохастической аппроксимации.—Методы решения задач негладкой оптимизации и стохастического программирования.—К., 1979, с. 18—25. (Препринт 79—75. Ин-т кибернетики АН УССР). 11. Ермолов Ю. М., Каниовский Ю. М. Асимптотические свойства некоторых методов стохастического программирования с постоянным шагом.—Журн. вычислит. математики, 1979, № 2, с. 356—366.

Поступила в редколлегию 20.06.80.

УДК 62.501

В. П. КОЛЬЦОВ, канд. техн. наук

РЕШЕНИЕ ОДНОГО КЛАССА ЗАДАЧ МЕТОДОМ ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Ранее* построена математическая модель задачи раскроя для специфичной группы промышленных материалов. Специфика выделенной группы материалов обусловлена их неоднородностью — параметр, описывающий свойства материала, является переменной величиной. Там же дан алгоритм решения задачи с помощью введенного структурного вектора λ .

Однако при увеличении числа сортов p этот алгоритм не может удовлетворить высоким требованиям к оперативности получения решения, так как, например уже при $p=5$ необходимо в общем случае 2^p раз применять алгоритм, полученный в работе*.

Используя введенные в упомянутой работе обозначения и терминологию, покажем, что можно построить алгоритм оптимального раскроя неоднородных промышленных материалов на основе метода динамического программирования.

В дальнейшем рассмотрим задачу раскроя не только отрезка $[0, L]$, но и любых его частей. Очевидно, некоторое подмножество точек любого раскройного плана $a = [a_1, \dots, a_n]$ отрезка

* Свиридов В. В., Илюмин С. К., Кольцов В. П. О задаче оптимального раскроя неоднородных промышленных материалов.—АСУ и приборы автоматизации, 1979, вып. 52, с. 92—97.

$[0, L]$ будет в свою очередь раскройным планом для отрезка $[l_1, l_2] \subset [0, L]$. Обозначим это следующим образом:

$$a[l_1, l_2] = a \cap [l_1, l_2].$$

Пусть $f(a[l_1, l_2])$ — значение функционала (8) (см. работу В. В. Свиридова и др.) при раскройном плане $a[l_1, l_2]$ отрезка $[l_1, l_2]$. В частности, при $l_1 = a_i$ и $l_2 = a_j$, $i < j$ введем обозначения:

$$a_{ij} = a[a_i, a_j], \quad f(a_{ij}) = f(a[a_i, a_j]).$$

Сформулируем утверждение, играющее для рассматриваемой задачи роль принципа максимума.

Утверждение 1. Если a^* — ОРП для отрезка $[l_1, l_2]$, то любое его подмножество a_{ij} является оптимальным раскройным планом для отрезка $[a_i, a_j]$.

Доказательство. Для любого раскройного плана a имеем

$$f(a[l_1, l_2]) = f(a_{ij}) + \phi, \quad (1)$$

где величина

$$\phi = f(a[l_1, a_i]) + f(a[a_j, l_2]) - r(a_j)$$

не зависит от $a_{i+1}, a_{i+2}, \dots, a_{j-1}$.

Пусть вопреки доказываемому раскройный план a_{ij} не является оптимальным для отрезка $[a_i, a_j]$. Тогда для этого отрезка существует раскройный план $\tilde{a}_{ij} = (\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_k)$ такой, что

$$f(\tilde{a}_{ij}) > f(a_{ij}). \quad (2)$$

Рассмотрим раскройный план

$$\tilde{a} = (a_1, \dots, a_i, \tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_k, a_j, \dots, a_n)$$

отрезка $[0, L]$. Поскольку величина ϕ не зависит от раскройного плана на отрезке $[a_i, a_j]$, то из (1) и (2) имеем

$$f(\tilde{a}[l_1, l_2]) - f(a^*[l_1, l_2]) = f(\tilde{a}_{ij}) - f(a_{ij}) > 0,$$

что противоречит оптимальности раскройного плана a^* . Утверждение 1 доказано.

Обозначим для любого $t \in [0, L]$ через a^* ОРП отрезка $[0, t]$, а $f^*(t)$ — значение функционала при этом, т. е. $f^*(t) = f(a^*[0, t])$.

Утверждение 2. Для любого $t \in [0, L]$ функционал $f^*(t)$ удовлетворяет функциональному уравнению

$$f^*(t) = \max \left\{ \sup_{0 < y < t-h} \{ f^*(y) + c(y, t)(t-y) - r(y) \}, \sup_{t-h < y < t} \{ f^*(y) + C_{y+1}(t-y) - r(y) \} \right\}. \quad (3)$$

Доказательство. Обозначим через A множество всех раскройных планов a отрезка $[0, L]$. Тогда, согласно определению,

$$f^*(t) = \sup_{a \in A} \{ f(a[0, t]) \}.$$

Для любого подмножества $K \subset A$ положим

$$\hat{f}_K(t) = \sup_{a \in K} \{f(a|0, t)\}.$$

Отнесем к классу K_1 все такие раскройные планы a , что $t - a_{j_s} > h$, и к классу K_2 все такие раскройные планы a , что $t - a_{j_s} < h$. Здесь $a_{j_s} = \max \{a_j / a_j < t\}$ — первый справа рез на отрезке $[0, t]$ при раскройном плане a . Тогда $A = K_1 \cup K_2$ и $K_1 \cap K_2 = \Phi$. Поэтому

$$f^*(t) = \sup_{a \in K_1 \cup K_2} f(a|0, t) = \max \left\{ \sup_{a \in K_1} f(a|0, t), \sup_{a \in K_2} f(a|0, t) \right\} = \max \{ \hat{f}_{K_1}(t), \hat{f}_{K_2}(t) \}. \quad (4)$$

Для $a \in K_1$

$$f = (a|0, t) = f(a|0, a_{j_s}) + c(a_{j_s}, t)(t - a_{j_s}) - r(a_{j_s}). \quad (5)$$

Для $a \in K_2$

$$f(a|0, t) = f(a|0, a_{j_s}) + c_{p+1}(t - a_{j_s}) - r(a_{j_s}). \quad (6)$$

Вычислим величину $\hat{f}_{K_1}(t)$. Заметим, что в силу утверждения 1 нахождение оптимума для величины $f(a|0, L)$ при фиксации первого правого реза состоит в оптимальном выборе раскройного плана $a|0, a_{j_s}$ на отрезке $[0, a_{j_s}]$. Поэтому

$$\begin{aligned} \hat{f}_{K_1}(t) &= \sup_{a_{j_s}/t - a_{j_s} > h} \left[\sup_{a|0, a_{j_s}} \{f(a|0, a_{j_s}) + c(a_{j_s}, t)(t - a_{j_s}) - \right. \\ &\quad \left. - r(a_{j_s})\} \right] = \sup_{a_{j_s}/t - a_{j_s} > h} \{f^*(a_{j_s}) + c(a_{j_s}, t)(t - a_{j_s}) - r(a_{j_s})\}. \end{aligned}$$

Полагая $y = a_{j_s}$, можно записать, что

$$\hat{f}_{K_1}(t) = \sup_{y/t - y > h} \{f^*(y) + c(y, t)(t - y) - r(y)\}. \quad (7)$$

Аналогично

$$\hat{f}_{K_2}(t) = \sup_{y/t - y < h} \{f^*(y) + c_{p+1}y - r(y)\}. \quad (8)$$

Комбинируя (7), (8) и (4), приходим к требуемому равенству (3). Утверждение 2 доказано.

Таким образом, функциональное уравнение (3) позволяет для нахождения оптимального раскройного плана использовать метод динамического программирования. Применение этого метода позволяет при увеличении размерности задачи получить решение в практически применяемое время.

Поступила в редколлегию 19.02.82.

ИМИТАЦИОННАЯ МОДЕЛЬ ФУНКЦИОНИРОВАНИЯ
СКОРОЙ МЕДИЦИНСКОЙ ПОМОЩИ

Одной из подсистем здравоохранения является скорая медицинская помощь (СМП). Функциями СМП являются оказание квалифицированной медицинской помощи на месте нахождения больного и в случае необходимости доставка его в стационар.

СМП представляет собой иерархическую систему. Основными структурными элементами ее являются центральный диспетчерский пункт (ЦДП), подстанция (ПС) и бригада скорой медицинской помощи (БСМП). Важнейшими показателями эффективности работы СМП считаются время ожидания обслуживания $T_{ож. обл.}$ и время обслуживания вызова $T_{обсл.}$. Оба эти показателя — случайные величины, поэтому в дальнейшем будем рассматривать их средние значения $\bar{T}_{ож. обл.}$, $\bar{T}_{обсл.}$ как оценки математических ожиданий соответствующих случайных величин.

В общем случае время обслуживания вызова

$$T_{обсл.} = \sum_{i=1}^4 t_i \quad (1)$$

где t_1 — время обработки вызова на ЦДП и ПС, включая время передачи его бригаде СМП; t_2 — время проезда бригады к месту вызова; t_3 — время медицинского обслуживания больного на месте вызова; t_4 — время доставки больного в стационар (если в этом нет необходимости — $t_4 = 0$).

Время ожидания обслуживания — составная часть времени обслуживания:

$$T_{ож. обл.} = \sum_{i=1}^n t_i \quad (2)$$

Главная цель управления подсистемой СМП — минимизация времени ожидания обслуживания, так как оно во многом определяет затраты и время на лечение больного, а иногда даже вероятность его выздоровления. Если учесть, что время ожидания обслуживания является невозрастающей функцией множества материальных и трудовых ресурсов R , то формально цель подсистемы можно записать

$$T_{ож. обл.} = F(R) \rightarrow \min. \quad (3)$$

Достижение цели (3) возможно двумя взаимосвязанными путями: увеличением ресурсов R ; повышением эффективности использования имеющихся ресурсов ($R = \text{fics}$). Так как ресурсы всегда ограничены, в реальных условиях (3) редуцируется к виду

$$P(T_{ож. обл.} > \alpha) \leq \beta, \quad (4)$$

где β — уровень обслуживания, представляющий собой вероятность того, что $T_{\text{ож. obs}}$ не превысит некоторую наперед заданную величину α . В этих условиях критерием эффективности достижения цели (4) служит $R \rightarrow \min$ (5). Ресурсом для подсистемы СМП являются бригады, поэтому задача состоит в минимизации их числа при удовлетворении ограничения (4).

Традиционные методы планирования сети учреждений здравоохранения, включая и СМП, основаны главным образом на нормативах потребности населения в различных видах медицинских услуг [1]. Например, при традиционных методах планирования количество бригад скорой медицинской помощи, обслуживающих некоторый участок города, определяется исходя из нормативной потребности населения в услугах СМП, рассчитанной по отношению к 1000 или 10000 человек. Естественно, что такой подход не может обеспечить решение поставленной выше задачи, так как не учитывается динамика происходящих событий, игнорируются элементы случайности, расчет показателей крайне обобщенный, без учета особенностей конкретного района (социально-демографическая структура населения, топология района и др.), дисциплина обслуживания вызовов неизменна.

Анализ составляющих времени обслуживания (1) показал, что основным резервом его сокращения и, соответственно, уменьшения времени ожидания обслуживания $T_{\text{ож. obs}}$, является минимизация времени проезда к месту вызова. Время проезда определяется удалением ВСМП в момент приема вызова от места нахождения больного. Удаление в свою очередь — функция размеров и конфигурации зоны обслуживания, интенсивности поступления вызовов, места расположения подстанции и количества бригад СМП, закрепленных за данной территорией.

Для решения поставленной оптимизационной задачи с учетом перечисленных выше факторов необходима математическая модель, адекватно описывающая процесс функционирования СМП. Эксперименты с такой моделью должны давать ответ «что будет, если», например, изменятся размеры зоны обслуживания, место расположения подстанций и т. д...

Решение такой задачи требует построения математической модели функционирования СМП как системы массового обслуживания (СМО). Специфической чертой СМП как системы массового обслуживания является то, что источники требований и обслуживающие устройства рассредоточены на некоторой территории.

В настоящее время при разработке моделей СМО используются два подхода: аналитические методы [2, 3]; методы имитационного моделирования [4, 5] аналитических методов является возможность получения результатов с высокой степенью точности. Однако введение в аналитическую модель новых

факторов (переменных) или зависимостей требует разработки нового метода решения задачи.

Более универсальное средство для описания СМО — имитационное моделирование на ЭВМ. К его преимуществам можно отнести: возможность учета случайных факторов и зависимостей между ними; получение в результате моделирования более содержательной информации о функционировании системы; простота изменения моделирующего алгоритма, т. е. имитационные модели легко адаптируются к качественным изменениям модели, в то время как аналитические — только к количественным. В связи с этим при разработке модели функционирования СМП нами использован второй подход, основанный на методе статистического моделирования.

При синтезе модели приняты следующие допущения.

1. Город представлен в виде ориентированного графа $G(X, \Gamma)$, где X — множество вершин графа, представляющих собой отдельные дома или группы домов (X_1) и перекрестки транспортной сети (X_2): $X = X_1 \cup X_2$ (6), Γ — множество путей, соединяющих эти вершины.

2. Для каждой i -й подстанции ($i = \overline{1, n}$) задано подмножество вершин $X_{i, k}$, образующих ее зону обслуживания. Причем

$$\bigcup_{i=1}^n X_{i, k} = X_{i, k},$$

где n — количество подстанций в городе. Место расположения подстанции задано и совпадает с одной из вершин множества $X_{i, k}$ ($k = 1, 2$).

3. На подстанцию поступают вызовы с интенсивностью

$$\lambda = \sum_{j=1}^m \lambda_j, \quad (8)$$

где m — количество вершин в зоне обслуживания; λ_j — интенсивность поступления вызовов из j -вершины.

4. Бригады СМП i -подстанции обслуживают вызовы только своей зоны обслуживания.

5. Обслуживание всех заявок происходит с одинаковым приоритетом (бесприоритетное обслуживание).

6. Закон распределения времени обслуживания больного врачом бригады СМП для всех вызовов одинаков.

Алгоритм моделирования предполагает следующую дисциплину обслуживания заявок. На подстанцию с ЦДП с некоторой интенсивностью поступает поток заявок на обслуживание. Предполагается, что в момент начала моделирования все бригады находятся на ПС. По мере поступления вызовов бригады выезжают на обслуживание. После обслуживания очередной заявки бригада возвращается на ПС. Следует отметить, что разработанный алгоритм учитывает и те случаи, когда вызов застает бригаду на пути следования ее от места обслуживания очередной

заявки на подстанцию. Если в момент поступления заявки все бригады заняты, заявка «становится» в очередь и «ожидает». Ее обслуживает первая освободившаяся бригада. Если в момент поступления заявки имеется несколько «свободных» бригад, то вызов обслуживает бригада, ближе других находящаяся к месту вызова. В случае, если необходима срочная госпитализация больного, бригада доставляет его в один из стационаров. При выборе стационара решающим является наличие в нем свободных мест, а в качестве критерия выступает минимум расстояния от места нахождения больного до стационара.

Моделирующий алгоритм построен по модульному принципу и имеет следующий вид:

$$A_1^{23} A_2^{19} \Phi_3 A_4 P_{5, 8} A_6 F_7 A_8 P_{9, 11} A_{10}^9 \Phi_{11} A_{12} P_{13, 15} \\ K_{14}^{13} A_{15} A_{16} K_{17} A_{18} P_{19, 3} A_{20} P_{21} P_{22, 24} K_{23}^2 P_{24} Y_{25},$$

где A_1 — начало моделирования; ввод исходной информации; A_2 — задание начальных условий; Φ_3 — формирование потока заявок; A_4 — определение номера вершины, из которой поступил вызов; P_5 — определение наличия «свободных» бригад; A_6 — определение времени «ожидания» бригадой очередного вызова; F_7 — формирование времени «ожидания» заявки в очереди; A_8 — поиск «свободной» бригады, ближе других расположенной к месту вызова; P_9 — определение места, с которого бригада выезжает на обслуживание очередной заявки; A_{10} — определение времени проезда бригады к месту очередного вызова; Φ_{11} — формирование времени медицинского обслуживания больного; A_{12} — определение времени ожидания обслуживания; P_{13} — проверка условия $T_{\text{ож. обслуж.}} > a$; K_{14} — счетчик заявок, для которых $T_{\text{ож. обслуж.}} > a$; A_{15} — определение времени проезда от места вызова до стационара; A_{16} — формирование времени освобождения ЕСМП после обслуживания очередного вызова; K_{17} — счетчик количества обслуженных заявок; A_{18} — определение суммарных затрат времени на обслуживание вызовов; P_{19} — проверка полученных результатов на точность; A_{20} — определение оценок параметров модели; P_{21} — печать промежуточных результатов; P_{22} — проверка выполнения ограничений; K_{23} — изменение количества бригад; P_{24} — печать окончательных результатов; Y_{25} — конец моделирования.

Алгоритм реализован на языке ФОРТРАН. Было промоделировано функционирование бригад скорой медицинской помощи трех зон обслуживания с различной топологией и интенсивностью поступления вызовов. Моделирование проводилось в предположении, что интервалы между поступлением соседних по времени заявок и время медицинского обслуживания распределены по показательному закону с параметрами λ и μ соответственно [2]. Модель позволяет легко изменить вид закона распределения потока заявок и времени медицинского обслуживания.

Интенсивность поступления вызова λ считаем постоянной во времени, так как сутки можно разбить на такие интервалы, на которых изменение интенсивности будет незначительным.

Построенная модель позволяет определить минимальное количество бригад СМП, необходимых для обеспечения в зоне с фиксированной топологией уровня обслуживания не ниже заданного. При этом учитываются социально-демографические особенности населения зоны обслуживания и ее топология.

Список литературы: 1. Попов Г. А. Экономика и планирование здравоохранения. — М.: Экономика, 1976. — 376 с. 2. Сестин Т. Л. Элементы теории массового обслуживания и ее приложения. — М.: Сов. радио, 1971. — 520 с. 3. Джейкобс Н. Очерки с приоритетами. — М.: Мир, 1973. — 279 с. 4. Бусленко Н. П. Моделирование сложных систем. — М.: Наука, 1978. — 399 с. 5. Баккас А. А. Имитационные модели в экономике. — К.: Наук. думка, 1978. — 303 с.

Поступила в редколлегию 20.04.81.

УДК 62—50

В. Д. ДОМРИН, канд. техн. наук, В. А. ШАПОВАЛЕНКО, Вад. И. САЛЫГА, канд. техн. наук

АЛГОРИТМ ОРТОГОНАЛИЗАЦИИ ФУНКЦИИ ПРИ ИДЕНТИФИКАЦИИ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ

Относительно простые алгоритмы идентификации динамических объектов могут быть получены предварительным сглаживанием, используемых при решении уравнений Винера — Хопфа, авто- и взаимно-корреляционных функций с последующим разложением решения по некоторой системе ортонормированных функций [1]. Известны ортогональные разложения с применением полиномов Чебышева, Эрмита, Лежандра, Лагерра, однако исследователи ищут новые функции, в лучшей степени удовлетворяющие решению задач выдвигаемых практикой. При этом возникают трудности расчета из известной системы функций ортонормированных. В настоящей работе предлагается алгоритм, позволяющий получить ортонормированные функции на ЭВМ.

Пусть задана система линейно независимых на интервале $(0, \infty)$ функций

$$f_1(t), f_2(t), \dots, f_n(t), \quad (1)$$

из которой требуется получить систему ортогональных функций

$$\varphi_1(t), \varphi_2(t), \dots, \varphi_n(t). \quad (2)$$

Ортогональная на этом интервале система (2) в соответствии с методом Грамма — Шмидта [2] (при $\varphi_1(t) = f_1(t)$) может быть определена уравнением

$$\varphi_i(t) = f_i(t) - \sum_{j=1}^{i-1} \frac{\langle f_i(t), \varphi_j(t) \rangle}{\langle \varphi_j(t), \varphi_j(t) \rangle} \varphi_j(t), \quad i = \overline{2, n}. \quad (3)$$

Однако определение $\varphi_i(t)$ по известным $f_i(t)$ в ряде случаев приводит уже при $i > 4$ к громоздким вычислениям.

Для определения закономерностей в уравнении (3) запишем его в векторно-матричной форме

$$\begin{bmatrix} 1 & 0 & 0 & 0 \dots \\ v_{21} & 1 & 0 & 0 \dots \\ v_{31} + v_{32}v_{21} & v_{32} & 1 & 0 \dots \\ v_{41} + v_{42}v_{21} + \\ + v_{43}(v_{31} + v_{32}v_{21}) & v_{42} + v_{43}v_{32} & v_{43} & 1 \dots \\ \dots & \dots & \dots & \dots \end{bmatrix} \begin{bmatrix} f_1(t) \\ f_2(t) \\ f_3(t) \\ \varphi_4(t) \\ \dots \end{bmatrix} = \begin{bmatrix} \varphi_1(t) \\ \varphi_2(t) \\ \varphi_3(t) \\ \varphi_4(t) \\ \dots \end{bmatrix}.$$

Полученную систему уравнений преобразуем к виду

$$\varphi_i(t) = \sum_{j=1}^i z_{ij} f_j(t), \quad i = \overline{1, n},$$

где коэффициенты ортогонализации определяются рекуррентным формулам

$$z_{ij} = \begin{cases} 1 & \text{при } i = j; \\ \sum_{k=j}^{i-1} v_{ik} z_{kj} & \text{при } i \neq j. \end{cases}$$

Здесь коэффициенты v_{ik} в соответствии с (3) имеют вид

$$v_{ik} = - \frac{\langle f_i(t), \varphi_k(t) \rangle}{\langle \varphi_k(t), \varphi_k(t) \rangle}. \quad (4)$$

Скалярные произведения двух функций в (4) могут быть определены через скалярные произведения исходных функций

$$\begin{aligned} \langle \varphi_k(t), \varphi_k(t) \rangle &= \sum_{i=1}^k z_{ki}^2 \langle f_i(t), f_i(t) \rangle + \\ &+ 2 \sum_{i=1}^{k-1} \sum_{m=i+1}^k z_{ki} z_{km} \langle f_i(t), f_m(t) \rangle, \\ \langle f_i(t), \varphi_k(t) \rangle &= \sum_{l=1}^k z_{kl} \langle f_i(t), f_l(t) \rangle. \end{aligned}$$

Таким образом, задача определения ортогональных функций (2) сводится к расчету коэффициентов z_{ij} . Для реализации алгоритма на ЭВМ необходимо знать скалярные произведения исходных функций $P_{lm} = \langle f_l(t), f_m(t) \rangle$, которые могут быть получены в общем виде для заданной системы (1).

Ортонормированные функции $\varphi_i^H(t)$ определяются уравнением

$$\varphi_i^H(t) = \varphi_i(t) / \sqrt{\langle \varphi_i(t), \varphi_i(t) \rangle},$$

где $\sqrt{\langle \varphi_i(t), \varphi_i(t) \rangle}$ — норма в ортогональной системе функций.

На рисунке приведена блок-схема алгоритма, реализованного на языке ФОРТРАН. Основные операторы алгоритма имеют следующие назначения.

1. Расчеты скалярных произведений исходных функций

$$P_{lm} = \langle f_l(t), \bar{f}_m(t) \rangle, \quad l = \overline{1, n-1}; \quad m = \overline{1, n}.$$

2. Присвоение z_{ii} значения единицы при $i = \overline{1, n}$.
3. Присвоение индексу i значение два.
4. Присвоение индексу j значение единицы.
5. Расчет коэффициентов

$$z_{ij} = z_{ij} - z_{ki} \times \frac{\sum_{l=1}^k z_{kl} P_{il}}{\sum_{l=1}^k z_{kl} P_{il} + 2 \sum_{l=1}^{k-1} \sum_{m=l+1}^k z_{kl} z_{km} P_{lm}},$$

$$k = \overline{j, i-1}.$$

6. Проверка равенства $i = 1 + j$.
7. Проверка равенства $i = n$.
8. Расчет ортонормированных коэффициентов

$$z_{ij}^n = z_{ij} \left(\sum_{l=1}^i z_{il} P_{il} + 2 \sum_{l=1}^{i-1} \sum_{m=l+1}^i z_{il} z_{im} P_{lm} \right)^{-1/2},$$

$$i = \overline{1, n}; \quad j = \overline{1, i}.$$

9. Прибавление единицы в счетчике индекса j .
10. Прибавление единицы в счетчике индекса i .

По окончании работы алгоритма имеем массив чисел элементов матрицы z_{ij}^n уравнения

$$\psi_i^n(t) = \sum_{j=1}^i z_{ij}^n f_j(t), \quad i = \overline{1, n}.$$

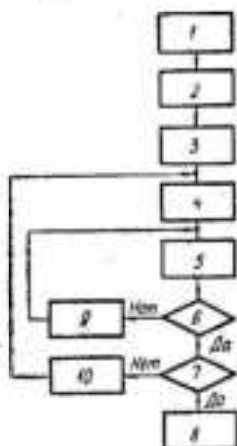
В качестве примера реализации алгоритма рассмотрим получение ортонормированных функций Каутца, используемых при идентификации технологических процессов [3].

Исходную систему функций (1) примем в виде

$$f_i(t) = \exp(-\alpha_i t),$$

для которой скалярные произведения двух функций определяются выражением

$$P_{lm} = \int_0^{\infty} \exp[-(\alpha_l + \alpha_m)t] dt = (\alpha_l + \alpha_m)^{-1}.$$



Блок-схема алгоритма, реализованного на ФОРТРАН

Данная информация и значение λ служат исходными для расчета коэффициентов z_{ij}^n на ЭВМ.

Расчет на ЭВМ для $n = 5$ и конкретных значений α_i позволил получить ортонормированные функции Каутца во временной области, преобразование Лапласа которых имеет вид

$$K_i(p) = \sqrt{2\alpha_i} \frac{\prod_{k=0}^{i-1} (p - \alpha_{ik})}{\prod_{k=1}^i (p + \alpha_{ik})}, \quad i = \overline{1, 5}.$$

Алгоритм является составной частью разрабатываемого математического обеспечения автоматизированных систем управления технологическими процессами.

Список литературы: 1. Кожинский О. С., Гроборь Л. П. Об одном методе идентификации линейных объектов управления. — Идентификация: Докл. II Всесоюз. совещ. по статист. проблемам управления. — М.: Наука, 1970, с. 39—50. 2. Фаддеев Д. К., Фаддеева В. Н. Вычислительные методы линейной алгебры. — М.: Л.: Физматгиз, 1963. — 734 с. 3. Салыга В. Н., Домрин В. Д., Шемша В. В., Штонда А. Н. Идентификация динамических характеристик объектов химической технологии. — Идентификация и оптимизация производственных комплексов. Препринт 76—5, Ин-т кибернетики АН УССР, К., 1975, с. 3—6.

Поступила в редколлегию 17.07.80.

УДК 007:658.012.011.56

К. Д. КОНОВАЛЕНКО, канд. экон. наук, Т. В. КОШЕЛЬ

СИНТЕЗ ОПТИМАЛЬНОЙ СТРУКТУРЫ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА ОБРАБОТКИ ИНФОРМАЦИИ О КАЧЕСТВЕ ПО ЦЕННОСТНОМУ КРИТЕРИЮ

Проектирование технологического процесса обработки информации о качестве (ТПОИ) на этапах проектирования, производства и эксплуатации изделия составляет основную часть работ по проектированию всей автоматизированной системы управления качеством продукции (АСУКП), включающей управление качеством НИР и ОКР. Степень рациональности ТПОИ о качестве определяет не только эффективность в сфере управления, но и качество продукции на всех стадиях ее жизненного цикла.

Цель работы — синтез оптимальной физической структуры технологического процесса обработки информации о качестве по ценностному критерию на основе метода, изложенного в [1].

ТПОИ о качестве оценивается по ряду показателей, которые определяются целью и сферой применения ТПОИ. Единственным информационным эквивалентом для всех показателей может служить функция ценности информации с аргументами, которые являются критериями ТПОИ. Как и раньше [2], понятие ценности инфор-

машии (V — качество) эксплицируется понятиями точность (ϕ — свойство), безошибочность обработки (φ — свойство), непротиворечивость (ψ — свойство), релевантность (ρ — свойство), полнота (h — свойство), краткость (ζ — свойство), наглядность (η — свойство), обозримость (θ — свойство), последовательность (ω — свойство), актуальность (r — свойство), оперативность (o — свойство), доступность (l — свойство), употребимость (u — свойство), новизна информации (n — свойство); таким образом,

$$V_{def} \phi \wedge \varphi \wedge \psi \wedge \rho \wedge h \wedge \zeta \wedge \eta \wedge \theta \wedge \omega \wedge r \wedge o \wedge l \wedge u \wedge n.$$

При синтезе ТПОИ о качестве по ценностному критерию использовались следующие утверждения.

Утверждение 1. Структура технологического процесса может быть логической A_L и физической A_F .

Утверждение 2. Логическая структура A_L ТПОИ задается упорядоченным множеством M операций m ($1, n$) по обработке оперативной и статистической информации о качестве.

Утверждение 3. Физическая структура ТПОИ строится по принципу связанных звеньев: технических (τ), организационных (z), антропологических (a); в каждом звене используется определенный носитель z информации.

Каждое отдельное свойство или совокупность свойств, перечисленных выше, проявляются в логической A_L и физической A_F структурах ТПОИ.

Физическая постановка задачи. Синтез физической A_F структуры ТПОИ осуществляется при заданной логической A_L структуре и заключается в выборе носителей z информации, способов F_H формирования полноты информации об исследовании процесса формирования качества, способов F_P подготовки информации для ввода в машину и методов F_D контроля достоверности информации о качестве проектирования и изготовления аппаратуры, которые обеспечат некоторые уровни показателей качества ТПОИ. В качестве последних приняты: 1) показатели ценности информации: достоверность V_D , оперативность V_O , актуальность V_R , полнота V_H информации; 2) затраты ручного труда на реализацию ТПОИ; 3) общие стоимостные затраты.

В такой постановке эти показатели измеряются в различных единицах, имеют несоизмеримые масштабы и взаимно противоречивы по своей природе. Поэтому синтез физической структуры технологического процесса обработки информации о качестве допускает отыскание компромиссных структур ТПОИ.

Значения показателей, мерой проявления которых являются не принятые в процедуре синтеза перечисленные выше свойства, эксплицирующие ценность информации, считаются заданными и служат ограничениями в постановке задачи.

Краткая характеристика объекта автоматизации. Объектом автоматизации являются процедуры накопления, хранения, обновления, поиска и выдачи информации о качестве изготовления

продукции. Автоматизированная реализация этих процедур составляет автоматизированную систему управления качеством изготовления продукции (АСУ КИП), которая входит в состав АСУ «Качество», являющейся в свою очередь подсистемой АСУП.

Объект автоматизации имеет следующие специфические особенности: нерегулярность формирования первичного документа о качестве (сигнального листа — СЛ) вследствие вероятностного характера возникновения отказов (дефектов) аппаратуры; открытие сигнального листа производится на всех уровнях изготовления изделия (системы, устройства, прибора, блока, узла, платы, ячейки); открытие сигнального листа производится на отказы различной категории: покупные электро-радио изделия (ЭРИ), производственные дефекты, дефекты изделий смежных цехов, самоустранившиеся отказы, единичная некорректная, технологические и конструктивные недоработки; стохастический характер продолжительности формирования сигнального листа, определяемый длительностью исследования отказа (дефекта); наличие нескольких источников при формировании сигнального листа; ввод информации об отказе (дефекте) в два этапа: на первом — информация об отказе, фиксирующая факт отказа; на втором — полная информация об отказе в результате законченного исследования причин отказа.

В АСУ КИП в результате автоматизации ТПОИ решаются два комплекса задач: автоматизированное формирование таблиц о качестве для главных специалистов и автоматизированное формирование ответов на запросы о качестве изготовления продукции. Первый комплекс задач решается каждый месяц, квартал, полугодие, год. В результате его решения за каждый период формируются 19 машинограмм. Второй комплекс задач характеризуется запросами о качестве изготовления продукции, которые могут быть детерминированными стационарными, детерминированными нестационарными и вероятностными. Следовательно, моменты автоматизированного формирования ответов на запросы носят аналогичный характер. ТПОИ, на основе которого решаются два комплекса задач, имеет 37 операций типа m . Из таблицы следует, что при проектировании ТПОИ рассматриваются 480 комбинационных структур.

Постановка задачи. Введем следующие обозначения: $K = \{k\} = T \cup \exists \cup A$ — множество звеньев (технических, эргатических и антропологических) физической структуры ТПОИ; $S = \{s\}$ — множество носителей информации, которые могут быть использованы звеньями « t_k », « $\exists_k$ », « a_k » при реализации любой m -й операции; $F_H = \{f_h\}$ — множество способов формирования полноты информации об исследовании отказов; $F_D = \{f_d\}$ — множество методов контроля, которые могут быть использованы звеньями « t_k », « $\exists_k$ », « a_k » при реализации m -й операции; $F_P = \{f_p\}$ — множество способов подготовки данных для ввода в ЭВМ; $P_B(m - x, k, f_d, P_B(m - y, k, f_d))$ — вероятность недостоверности вход-

Машинный		Ручной		Первичный документ
Факт и причина отказа	Факт отказа	Факт и причина отказа	Факт отказа	Заключенность исследования
—		Формирование СЛ Формирование ОС		Тип ручного документа
Формирование МСЛ		Формирование МКУ01 на основании СЛ Формирование МКУ01 на основании ОС		База для заполнения М/О документа
Формирование МОС				
Визуальный Логический				Вид контроля
МСЛ МОС		МКУ01		Тип документа, поступающего на ГВЦ
Устройство подготовки данных на перфокартах (УПДК) Устройство подготовки данных на магнитную ленту (УПДМК)				Тип устройства подготовки данных
Визуальный Логический Программный				Вид контроля
	Дополнение машинно-ориентированного сигнального листа	— —	Формирование МКУ01 на основании закрытого СЛ Дополнение МКУ01 на основании закрытого СЛ	Регистрация первичной информации об окончательном исследовании отказа
	Визуальный Логический Программный	—	Визуальный Логический Программный	Вид контроля
	УПДК УПДМЛ	—	УПДК УПДМЛ	Тип устройства подготовки данных
2 ¹ · 3 = 24	2 ¹ · 3 ² = 144	2 ³ · 3 = 24	2 ⁶ · 3 ² = 288	Количество вариантов ТПОИ

ной и выходной информации о качестве после контроля информации f_d -методом соответственно на x -выходе и y -выходе m -й операции k -го звена; $\tau_1(m, k, i)$ — момент, фиксирующий начало проведения m -й логической операции с k -м методом контроля для i -го запроса из комплекса $B = \{b_i\}$; $\tau_2(m, k, i)$ — момент, фиксирующий полную готовность использования в m -й логической процедуре с k -м методом контроля для i -го запроса из комплекса $B = \{b_i\}$; $j = \overline{1, p}$ — множество возможных значений момента; B_V — число запросов, при удовлетворении которых не выполняется одно из требований к ценности информации (оперативность);

$$x_k^m = \begin{cases} 1, & \text{если в } m\text{-й операции участвует } k\text{-е звено;} \\ 0 & \text{в противном случае;} \end{cases}$$

$$z = \begin{cases} 1, & \text{если } B_V = 0; \\ 0, & \text{если } B_V = \text{const;} \end{cases}$$

$\tau_a(i)$ — момент наблюдения явления (отказа) — источника первичной информации, которая используется в i -м запросе; P_{HD} , P_{HA} — критерий дифференцированной и агрегированной полноты информации об отказе соответственно; n_a — количество отказов (дефектов), информация о которых используется при удовлетворении запроса, КИП; n_0 — общее число отказов (дефектов), обнаруженных до начала удовлетворения поступившего запроса о КИП; r — число реквизитов об отказе, зафиксированных для ввода в ЭВМ; r_0 — общее число реквизитов, заполняемых при обнаружении и исследовании отказа (дефекта); σ_k^m — трудоемкость ручной операции m в звене k : «э_к» или «а_к»; c_k^m — стоимость звена k , используемого в операции m : для «а_к» стоимость есть цена — $c^m(\tau_k)$; для «э_к» стоимость определяется заработной платой лиц, реализующих звено, — $c^m(a_k)$; для «э_к» — $c^m(\tau_k) + c^m(a_k)$; c_k^d — стоимость f_d -го метода контроля в k -м звене.

В общем случае задача многокритериальной оптимизации физической структуры ТПОИ о качестве формулируется следующим образом.

Дано: 1) множества B запросов на информацию о качестве; 2) логическая A_L структура ТПОИ, определяемая множеством M логических операций; 3) множества K, S, F_H, F_D, F_P .

Требуется: найти такие подмножества $\bar{K} \subset K, \bar{S} \subset S,$

$$\bar{F}_H \subset F_H, \bar{F}_D \subset F_D, \bar{F}_P \subset F_P,$$

чтобы

$$V_D = 1 - \prod [P_D(m-x, k, i), P_D(m-y, k, i)]^{x_k^i y_k^i} \rightarrow \max, \quad (1)$$

$$V_A = \frac{\inf_i \tau_1(m, k, i) - \sup_i \inf_i \tau_2(i)}{\inf_i \tau_1(m, k, i) - \tau_2^0(i)} \rightarrow \max, \quad (2)$$

$$V_0 = \frac{\sum_k \sum_l [\inf_i \tau_l(m, k, i) - \tau_k^0(m, k, i)] x_k^m}{\sum_k \sum_l [\inf_i \tau_l(m, k, i) - \sup_i \inf_i \tau_k(m, k, i)] x_k^m} \left(z + \frac{B - B_0}{B} \right) \rightarrow \max \quad (3)$$

$$V_H = P_{HD} \cdot P_{HA} = \frac{n_a}{n_0} \cdot \frac{r}{r_0} \rightarrow \max, \quad (4)$$

$$V_s = \sum_k \sum_m \sigma_k^m X_k^m \rightarrow \min, \quad (5), \quad V_C = \sum_k \sum_m (c_k^m + d_k^m) x_k^m x_k^m \rightarrow \min. \quad (6)$$

при условиях практической реализации ТПОИ о качестве на множествах: $\hat{K}, \hat{S}, \hat{F}_H, \hat{F}_D, \hat{F}_P$.

В постановке (1) — (6) технологический процесс обработки информации о качестве рассматривается расчлененным на множество операций типа m . Качество ТПОИ на каждой операции оценивается локальным векторным критерием и на множестве операций — обобщенным векторным критерием. Каждый из критериев, входящих в локальный векторный критерий, зависит от условий, при которых будет реализовываться любая операция типа m и, следовательно, компоненты локального векторного критерия также являются векторами. Таким образом, в данной постановке задача синтеза оптимальной структуры ТПОИ является шестимерной, разномасштабной задачей трехвекторной оптимизации.

Алгоритм МО (многокритериальная оптимизация).

МО1. Определение области допустимых решений. Область допустимых решений реализуется в виде избыточной физической структуры ТПОИ (таблица).

МО2. Определение вариантов физической структуры. Варианты физической структуры ТПОИ определяются путем анализа и расчленения избыточной физической структуры.

МО3. Трехвекторная оценка вариантов структур. Каждая физическая структура оценивается по критериям (1) — (6).

МО4. Определение компромиссных структур. Процедура осуществляется на основе построения матрицы эффективности.

МО5. Построение схемы нормализации критериев. Нормализация критериев осуществляется по идеальному вектору эффективности ТПОИ в виде $V_E = (\max V_D, \max V_O, \max V_R, \max V_H, \min V_C, \min V_s)$.

МО6. Построение нормализованного пространства критериев. Нормализованное пространство строится на основе нормализованных критериев в виде относительных: $V_{EH} = \frac{V_E}{\max V_E}$ — при условии, что улучшение эффективности ТПОИ связано с максимизацией критерия; $V_{EH} = \frac{\min V_E}{V_E}$ — при условии, что улучшение эффективности ТПОИ связано с минимизацией критерия.

МО7. Построение схемы компромисса. Отыскание оптимального вектора эффективности ТПОИ основывается на определении минимального отклонения от идеального вектора, т. е.

$$\text{opt } V = \min \sum_{i=1}^6 (1 - V_{in}).$$

Описанный способ оптимизации физической структуры ТПОИ реализован в АСУ качеством изготовления продукции на одном из приборостроительных заводов г. Харькова.

Список литературы: 1. Ашеров А. Т. Синтез структур технологических процессов обработки информации с учетом показателей надежности. — Управляющие системы и машины, 1978, № 3, с. 10—15. 2. Ашеров А. Т., Кошель Т. В. Об уточнении понятия «ценность экономической информации». — АСУ и приборы автоматики, 1979, вып. 48, с. 11—16.

Поступила в редколлегию 06.04.81.

УДК 681.5

А. Н. ГВОЗДИНСКИЙ, канд. техн. наук, В. Ф. ДЗЮБЕНКО, В. М. ШПЕНЕВ

ОЦЕНКА ЭФФЕКТИВНОСТИ СИСТЕМ КОММУНИКАЦИИ РЕСУРСОВ ОПЕРАЦИОННОЙ СИСТЕМЫ ЭВМ В АСУ

Операционная система управляет четырьмя основными ресурсами: памятью, процессорами, устройствами и информацией (табл. 1) [1].

Таблица 1

Ресурсы	Примеры	Примеры соответствующего программного обеспечения
Память	Основная память	Программы управления памятью и страничной организации
Процессоры	Центральные процессоры каналов ввода-вывода	Регулирующие и планирующие
Устройства	Ленты, барабаны, перфораторы карт	Программы системного вывода (SPOOLING)
Информация	Сегменты (системные, пользователя, библиотечные)	Файловые системы, программы ведения библиотек и управления сегментами

В настоящее время можно говорить об использовании общей теории управления ресурсами в теории операционных систем, в рамках которой единые теоретические положения и методы оказываются применимыми к управлению памятью, процессорами, устройствами, информацией.

Эффективность коммуникаций ресурсов операционной системы оценивается с помощью четырех экспоненциальных составляющих эффективности, вычисляемых как мера производительности системы для каждой элементарной операции [2].

Система коммуникаций ресурсов операционной системы может быть представлена в матричном виде или с помощью графов, причем рабочие ресурсы представляются точками, а коммуникационные пути — связями (линиями) между точками. Если A , B и C — точки и если корреспондент A может посылать сообщения в B и C , а корреспондент B — только в A , то такая ситуация может быть представлена с помощью приводимого на

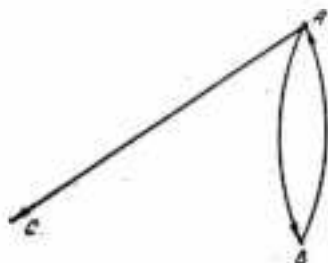


Рис. 1. Граф и соответствующая ему матрица

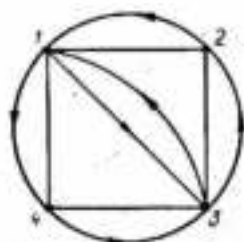


Рис. 2. Коммуникационная система ресурсов операционной системы

рис. 1 графа и соответствующей ему матрицы, элементы которой (единицы) указывают на связь в направлении от точки, соответствующей букве, помещенной слева, к точке, соответствующей букве, помещенной сверху матрицы (7).

Необработанные данные, используемые при оценке эффективности коммуникаций ресурсов операционной системы, представляются в форме матрицы коммуникаций типов ресурсов. Первоначально вычислительная машина формирует из матрицы коммуникаций квадратную подматрицу таким образом, чтобы она состояла только из тех строк и столбцов, которые представляют рабочие типы ресурсов, содержащие элементы ресурсов, используемые соответствующим пользователем во время выполнения соответствующей операции. Если x_n — число таких рабочих ресурсов, задействованных при выполнении элементарной операции n , подматрица имеет размерность $x_n \times x_n$. Затем вычисляются четыре компонента для получения E_{00} .

Компонент масштабируемости. Вычисляется для оценки ухудшения эффективности коммуникаций с увеличением масштаба задействованной в элементарной операции подсистемы, т. е. числа разновидностей задействованных ресурсов. Его максимальная величина предполагается равной единице для минимальной подсистемы, состоящей из двух разновидностей ресурсов, и сни-

жается с увеличением числа задействованных ресурсов. Величина масштабного компонента равна

$$\exp \left[- \left(1 - \frac{2}{x_u} \right) \right], \quad (1)$$

где x_u — число разновидностей ресурсов, задействованных при выполнении операции u .

Компонент связности. Он должен отражать повышение эффективности коммуникаций при большом числе взаимных связей в системе. Полной системой коммуникаций называют такую, в которой обеспечена связь по всем направлениям между всеми разновидностями ресурсов. Поскольку в этом случае для x_u задействованных разновидностей ресурсов имеется $x_u(x_u - 1)$ возможных каналов связи, рассматриваемый компонент может быть оценен следующим образом:

$$\exp \left\{ - \left[1 - \frac{l_u}{x_u(x_u - 1)} \right] \right\}, \quad (2)$$

где l_u — общее число связей, т. е. число единиц в подматрице коммуникаций.

Компонент равномерности нагрузки. Сумма элементов в некоторой строке подматрицы соответствует числу каналов связи, подходящих к рассматриваемой разновидности ресурсов. В идеале при равномерной нагрузке все суммы по строкам и столбцам должны быть одинаковыми. Неравномерность является функцией величины

$$\sum_{x_u} |S_r - \bar{S}| \text{ и } \sum_{x_u} |S_c - \bar{S}|, \quad (3)$$

где S_r — сумма единиц в строке $r = \overline{1, x_u}$; S_c — сумма единиц в столбце $c = \overline{1, x_u}$, а $\bar{S}$ — среднее число единиц в строках или столбцах, $\bar{S} = l_u/x_u$. Компонент равномерности, имеющий максимальное значение 1, оценивается как

$$\exp \left\{ - \frac{\sum_{x_u} |S_r - \bar{S}| + \sum_{x_u} |S_c - \bar{S}|}{2l_u} \right\}. \quad (4)$$

Компонент изолированности. Характеризует изолированность задействованных разновидностей ресурсов, определяет общее число разновидностей ресурсов, которые не имеют либо входящих, либо отходящих каналов связи, т. е. лишены возможности общения по крайней мере в одном направлении. Оценивается как

$$\exp \left\{ - \frac{v_c + v_r}{2x_u} \right\}, \quad (5)$$

где v_c равно числу пустых столбцов, v_r — числу пустых строк в подматрице.

Модель использует все четыре компонента с одинаковым весом и дополнительно рассматривает пятый, учитывающий помехи в системе коммуникаций. Помехи считаются распределенными по нормальному закону. Поэтому логично сначала вычислить среднее значение эффективности коммуникаций по четырем рассмотренным компонентам, а затем для учета помех определять реализуемое значение E_{0e} для каждой элементарной операции, умножая полученное среднее на число, выбранное из нормального распределения. Математическое ожидание этого компонента равно 0,9, среднее квадратическое отклонение 0,03, а сверху он ограничивается значением 1,0. Таким образом, помехи снижают эффективность коммуникаций в среднем до 0,9 с возможным изменением в пределах от 0,81 до 0,99 (три сигмы) в 99,7 % случаев. Тогда полная эффективность коммуникаций ресурсов операционной системы при выполнении элементарной операции подсчитывается по формуле:

$$E_{0e} = \left\{ \frac{1}{4} \exp \left[- \left(1 - \frac{2}{x_u} \right) \right] + \exp \left[- \left(1 - \frac{I_u}{x_u (x_u - 1)} \right) \right] + \right. \\ \left. + \exp \left[- \frac{\sum |s_r - \bar{s}| + \sum |s_e - \bar{s}|}{2J_u} \right] + \right. \\ \left. + \exp \left[- \left(\frac{v_e + v_r}{2x_u} \right) \right] \right\} (0,9 + 0,03R_u), \quad (6)$$

где R_u — нормально распределенное случайное отклонение с диапазоном значений вида 0,XXX.

С учетом сказанного коммуникационная система ресурсов операционной системы будет иметь такой вид (табл. 2).

Таблица 2

Константы					Компоненты			Полная эффективность операционной системы
x_u	v_e	v_r	I_u	масштабности	связности	равномерности нагрузки	напряженности	
4	0	0	10	0,61	0,85	1	1	0,79

В табл. 2 и на рис. 2 в конфигурации системы нумерации вершин графа 1, 2, 3, 4 соответствует номеру вида ресурса операционной системы, приведенной в табл. 1.

Операционная система рассмотрена как управляющая система множеством ресурсных групп.

В целях разработки метода повышения эффективности вычислительного процесса в АСУ представим программы, обслуживаемые системой, процессами, являющимися совокупностями данных. Имеются четыре ресурсные группы, которыми управляет операционная система [1]. Каждая из этих групп содержит

определенное число параллельно работающих идентичных установок. Опишем систему с «точки зрения» ресурсной установки.

Программы, обслуживаемые системой, представляются процессами, которые являются пассивными совокупностями данных. Эти программы состоят из заказов, каждый из которых содержит заданное количество предметов одинакового типа. Число типов программ задано. Каждый тип характеризуется определенным способом и заданным временем обработки.

Для каждой группы ресурсных установок (номер ГУ) существует набор НАЛИЧНЫЕ [ГУ], состоящий из незанятых установок, и набор ОЧЕРЕДЬ [ГУ], который представляет собой очередь программ, подлежащих обработке одного пакета программ. Каждый пакет включает заданное количество предметов, которые должны принадлежать одному и тому же заказу. Размер пакета зависит от типа продукции и ресурсной группы установок.

Очередь программ рассматривается как очередь заказов. Движение в очереди происходит по правилу: «первый вошел — первый вышел», при этом местоположение заказа в очереди определяется прибытием первого предмета этого заказа. Однако, если предметы данного заказа, ожидающего очереди, не составляют одного пакета, то рассматривается следующий заказ. Последние предметы заказа воспринимаются как отдельный пакет, даже если их количество меньше размера обычного пакета.

Если ресурсная установка не находит приемлемого пакета в очереди программ, то она будет ожидать дополнительного поступления.

Хотя программы состоят из предметов, последние не рассматриваются как индивидуальные единицы в настоящей модели. Для данного заказа и заданного шага в графике его выполнения, т. е. в работе ресурсной группы установок, мы определяем часть заказа — группу предметов, связанных в текущий момент с этим шагом. Предметы либо подвергаются обработке, либо ожидают ее начала у соответствующей ресурсной группы установок.

Заказ представляется совокупностью частей (заказов). Сумма предметов в каждой такой части равна количеству предметов в заказе. Каждая часть заказа является членом очереди программ. Если в плане выполнения какого-либо типа программ одна и та же ресурсная группа установок встречается несколько раз, то в очереди программ к этой группе установок может оказаться несколько частей того же самого заказа. Часть заказа имеет следующие целочисленные атрибуты: номер заказа НОЗ, ТИП программ, ШАГ, количество ожидающих предметов КПЖ и количество предметов, находящихся в обработке КПО. Движение предметов в системе приводит к изменению значения атрибутов КПЖ и КПО.

Часть заказа возникает в тот момент, когда первый пакет заказа поступает на ресурсную группу установок. Она вносится в конец соответствующей очереди программ. Часть заказа остается членом этой очереди до тех пор, пока не начнут обрабатываться последние ее предметы. И эта часть заказа исчезнет из системы только тогда, когда закончится обработка последних ее предметов. Булевский атрибут ПОСЛЕДНИЙ необходим для того чтобы указывать, содержит ли данная часть заказа последние предметы заказа, участвующие в данном шаге.

В любой заданный момент времени предметы заказа могут быть распределены между несколькими ресурсными группами установок. И для каждой из них появится соответствующая часть заказа. Процесс, соответствующий каждой части заказа, ссылается на процесс, связанный со следующим шагом, через элементарный атрибут ПРЕЕМНИК. Следовательно, заказ представляется простой цепью из своих частей. «Голова» не имеет преемника, а у записи в «хвосте» атрибут «последний» равен true. «Перемещение» заказа по системе означает добавление к цепи новых «голов» и удаление «хвостов».

Три последовательных шага в расписании обработки программы операционной системы. Очередь программ состоит из частей заказа (кружки), соединенных вертикальными линиями (рис. 1). Части заказа, принадлежащие тому же самому заказу, соединяются горизонтальными линиями.

Установки представляются прямоугольниками [3].

		к		
		А	В	С
от	А	0	1	1
	В	1	0	0
	С	0	0	0

(7)

Пунктирная линия между частью заказа и установкой показывает пакет единиц, находящихся в обработке.

Когда пакет третьей части заказа в ОЧЕРЕДИ заканчивается, создается новая часть заказа, получающая этот пакет; эта часть заказа включается в ОЧЕРЕДЬ [К].

Классы выходных данных [3].

1. Текущие сообщения о работе.
 - 1.1. Длины очередей программ (число ожидающих единиц).
 - 1.2. Общее количество единиц в системе.
 - 1.3. Число незанятых установок в каждой группе и т. д.
2. Накопленные статистические данные по установкам.
3. Накопление статистических данных по заказам.

Список литературы. 1. Функциональная структура OS/360/. Пер. с англ. под ред. В. С. Штаркмана. — М.: Сов. радио, 1971. — 88 с. 2. Зидель А. Вольф. Дж. Модели группового поведения в системе человек — машина. — М.: Мир, 1973. — 261 с. 3. Языки программирования/ Редактор Ф. Женюа. — М.: Мир, 1972. — 406 с.

Поступила в редколлегию 29.04.80.

ИССЛЕДОВАНИЕ СИСТЕМЫ АКСИОМ АЛГЕБРЫ КОНЕЧНЫХ ПРЕДИКАТОВ

В работах [1, 2] предложена алгебра конечных предикатов — удобный математический аппарат для описания конечных информационных процессов. Эта алгебра является обобщением аппарата булевых функций (переход от двоичных знаков к буквам) и аппарата многозначной логики (переход от функций к отношениям).

В данной статье исходя из практических соображений строится и анализируется минимальная система аксиом алгебры конечных предикатов.

Подобное исследование обычно проводят в два этапа. На первом этапе производят элиминацию аксиом, выводимых из остальных, а на втором доказывают независимость оставшихся после элиминации аксиом. Поскольку большая часть аксиом алгебры конечных предикатов совпадает с аксиомами булевой алгебры, то на первом этапе воспользуемся результатами, полученными при анализе систем аксиом булевой алгебры [3]. Следует отметить, что решая задачу нахождения минимальной системы аксиом алгебры конечных предикатов удалось построить экономную, симметричную систему аксиом и для булевой алгебры.

Материал разбит на три пункта: 1 — изложение системы аксиом алгебры конечных предикатов, 2 — элиминация выводимых аксиом, 3 — доказательство независимости оставшихся аксиом.

1. Система аксиом алгебры конечных предикатов. При изложении аксиом алгебры конечных предикатов мы не будем касаться ее содержательной интерпретации, а опишем ее формально.

Элементарным объектом этой алгебры является «узнавание» x^i , $i = 1, 2, \dots, k$, а операциями — «+» и «.» (последний знак можно опускать). Элементы x^i принимают значения 0 или 1, что управляется следующими аксиомами: $\omega_1: x^1 + x^2 + \dots + x^k = 1$, $\omega_2: \forall i, x^i \cdot x^i = 0$.

Основные элементы x^i соединяются в формулы $A, B, C, \dots$ по следующим рекурсивным правилам: 1) константы 0 и 1 — формулы; 2) элементы x^i — формула; 3) если A и B — формулы, то $(A + B)$ и (AB) — формулы.

Действия над всеми формулами $A, B, C, \dots$ подчиняются следующим аксиомам:

1. Коммутативные законы — $B_1: A + B = B + A$, $B_1': AB = BA$.
2. Ассоциативные законы — $B_2: (A + B) + C = A + (B + C)$, $B_2': (AB)C = A(BC)$, (т. е. можно писать без скобок): $A + B + C$, ABC .

3. Дистрибутивные законы — $B_3: (A+B)C = AC + BC$, $B'_3: AB + AC = (A+B)C$.

4. Законы идемпотентности — $B_4: A + A = A$, $B'_4: AA = A$.

5. Законы поглощения — $B_5: A + AB = A$, $B'_5: A(A+B) = A$.

6. Свойства 0 и 1 — $B_6: A + 0 = A$, $B'_6: A1 = A$, $B_7: A + 1 = 1$, $B'_7: A0 = 0$.

Эта система аксиом полна в том смысле, что с ее помощью можно доказать тождественность любых двух формул, описывающих один и тот же конечный предикат (алгебра и строилась для описания конечных предикатов). Однако для удобства в ней вводится еще одна операция — отрицание « $\bar{}$ ». Эта операция определяется следующими аксиомами:

$$\text{Ш}_3: \bar{x^i} = x^1 + x^2 + \dots + x^{i-1} + x^{i+1} + \dots + x^k.$$

7. Инволютивность операции — $B_8: \bar{\bar{1}} = 0$, $B'_8: \bar{\bar{0}} = 1$.

8. Законы де-Моргана — $B_9: \overline{A+B} = \bar{A}\bar{B}$, $B'_9: \overline{AB} = \bar{A} + \bar{B}$.

9. Закон исключенного третьего — $B_{10}: A + \bar{A} = 1$.

10. Закон противоречия — $B'_{10}: A\bar{A} = 0$.

Докажем, что так определенная операция отрицания корректна, т. е. что для любого элемента $E \in \Sigma$ (Σ — совокупность всех элементов алгебры, т. е. констант 0, 1, узнаваний x_i и формул $A, B, C, \dots$) существует $D \in \Sigma$ такой, что $D = \bar{E}$, и такой элемент D — единственный.

Существование элемента D устанавливается просто. Если $E \in \{0, 1\}$, то D задается инволютивными законами; если $E \in \{x^1, x^2, \dots, x^k\}$, то D задается аксиомой Ш₃, если E — формула, то, многократно применяя закон де-Моргана, можно привести формулу к виду, когда знаки отрицания будут стоять только над отдельными узнаваниями x^i и константами, затем убрать все черточки по аксиомам Ш₃, B₃ и B'₃, что и дает окончательный вид элемента D . Нужно доказать, что в этой процедуре любой порядок действий приведет к одному и тому же объекту D .

Допустим противное, т. е. что $\bar{E} = D_1$ и $\bar{E} = D_2$ и т. д. Тогда по B₁₀, B'₁₀

$$E + D_1 = 1, E + D_2 = 1, ED_1 = 0, ED_2 = 0. \quad (1)$$

Воспользуемся преобразованиями, которые допускают наши аксиомы (номер аксиомы пишется под знаком равенства):

$$\begin{aligned} D_1 &= D_1 \cdot 1 = D_1 \underset{B_6}{(E + D_2)} = \underset{B'_1}{(E + D_2)} D_1 = \underset{B_4}{ED_1} + \\ &+ \underset{(1)}{D_2 D_1} = 0 + \underset{B_4}{D_2 D_1} = \underset{B_4}{D_2 D_1} = \underset{B'_1}{D_2 D_1}. \end{aligned}$$

$$\begin{aligned}
 D_2 &= D_2 \cdot 1 = D_2 (E + D_1) = (E + D_1) D_2 = ED_2 + D_1 D_2 = \\
 &= 0 + D_1 D_2 = D_1 D_2 + 0 = D_1 D_2.
 \end{aligned}$$

Итак, $D_2 = D_1 D_2 = D_1$, отсюда $D_1 = D_2$. Единственность обратного элемента доказана.

В приведенной выше системе аксиом S через $\Pi_1 - \Pi_3$ обозначены специфические аксиомы алгебры конечных предикатов, а через $B_1, B'_1 - B_{10}, B'_{10}$ — аксиомы булевой алгебры. Аксиомы булевой алгебры двойственны: из аксиомы B_k ($k = \overline{1, 10}$) аксиоме B'_k получаем заменой $+$ на $\cdot$ и обратно, а если в ней есть логические константы, то заменой 0 на 1 и обратно. Это свойство облегчит дальнейшую работу.

Система S содержит 23 аксиомы. Многие из них выводимы из остальных. Так, например, выводим аксиомы B_{10}, B'_{10} или B_2 . Но как раз эти аксиомы мы оставим, что позволит вывести многие другие.

2. *Элиминация аксиомы.* Изложение в этом пункте представляет собой ряд предложений, доказывающих правомерность элиминации одной-двух аксиом. Доказательства предложений отделяются знаками $\blacktriangleleft$ (начало) и $\blacktriangleright$ (конец) доказательства.

Предложение 1. Аксиома Π_2 выводима из системы $S_1 = S/\Pi_2 \blacktriangleleft$. Для доказательства предложения, понадобится доказать вспомогательное утверждение $\bar{a} = a$; $\bar{a} \cdot \bar{a} = \bar{a}$; $\bar{a} + \bar{a} = \bar{1} = 0$. $\bar{a} +$

$$\bar{a} = \bar{a} + \bar{a} = \bar{a} \cdot \bar{a} = \bar{0} = 1.$$

Таким образом, элемент $\bar{a}$, умноженный на $\bar{a}$, дает 0, а сложенный с $\bar{a}$ дает 1. По определению операции отрицания такой элемент существует и только один — это a . Следовательно, $\bar{a} = a$;

$$\begin{aligned}
 0 &= \bar{1} = \overline{x^1 + \dots + x^i + \dots + x^j + \dots + x^k} = \\
 &= \overline{x^1 + x^1 + x^2 + x^2 + \dots + x^{i-1} + x^{i-1} + x^i + x^{i+1} + x^{i+1} +} \\
 &+ \overline{\dots + x^{i-1} + x^{j-1} + x^j + x^{j+1} + x^{j+1} + \dots + x^k + x^k} = \\
 &= \overline{(x^1 + \dots + x^{i-1} + x^{i+1} + \dots + x^k) + (x^1 + \dots + x^{j-1} + x^{j+1} +} \\
 &+ \dots + x^k)} = \overline{x^i + x^j} = \bar{x^i} \cdot \bar{x^j} = x^i \cdot x^j \blacktriangleright
 \end{aligned}$$

Предложение 2. Аксиома Π_1 выводима из системы $S_2 = S_1 \setminus \Pi_1$;

$$\begin{aligned} \leftarrow x^i + \bar{x}^i & \underset{\Pi_1}{=} x^i + x^i + x^2 + \dots + x^{i-1} + x^{i+1} + \dots + x^k \underset{B_1}{=} \\ & = x^i + \dots + x^k. \end{aligned}$$

С другой стороны, $x^i + \bar{x}^i \underset{B_{10}}{=} 1$

Комбинируя эти два равенства, имеем Π_1

$$x^1 + x^2 + \dots + x^k = 1 \blacktriangleright$$

Предложение 3. Каждая из аксиом B_8 и B'_8 выводима из системы $S_3 = S_2 \setminus B_8, B'_8$.

$$\leftarrow \bar{1} = \bar{1} \cdot 1 = 0, \quad \bar{0} = \bar{0} + 0 = 1 \blacktriangleright$$

$\underset{B'_8}{\bar{1}} \quad \underset{B'_{10}}{\bar{0}} \quad \underset{B_8}{0} \quad \underset{B_{10}}{1}$

Вторая выкладка в доказательстве предложения 3 двойственна к первой: единицы заменяются нулями и наоборот, знак $+$ заменяется знаком $\cdot$ и наоборот, под знаком равенства ставятся штрихованные номера аксиом вместо нештрихованных и наоборот. Поэтому из пары двойственных аксиом впредь можно доказывать только одну, — разумеется, при условии, что в ходе вывода используются только аксиомы, имеющие в оставшейся системе свою «пару».

Предложение 4. Каждый из законов де-Моргана выводим из системы $S_4 = S_3 \setminus B_9, B'_9$.

$$\begin{aligned} \leftarrow (A + B) + \bar{A}\bar{B} & \underset{B_1}{=} \bar{A}\bar{B} + (A + B) \underset{B'_3}{=} (\bar{A} + (A + B)) (\bar{B} + (A + B)) = \\ & \underset{B_1, B_2}{=} ((A + \bar{A}) + B) ((B + \bar{B}) + A) \underset{B_{10}}{=} (1 + B) (1 + A) \underset{B_1}{=} \\ & \underset{B_1}{=} (B + 1) (A + 1) \underset{B_1}{=} 1 \cdot 1 = 1; \end{aligned} \quad (2)$$

$$\begin{aligned} (A + B) \cdot (\bar{A} \cdot \bar{B}) & \underset{B_4}{=} A (\bar{A}\bar{B}) + B (\bar{A}\bar{B}) \underset{B'_1 B'_2}{=} \bar{B} (A\bar{A}) + \bar{A} (B\bar{B}) = \\ & \underset{B'_{10}}{=} \bar{B} 0 + \bar{A} 0 \underset{B'_7}{=} 0 + 0 = 0. \end{aligned} \quad (3)$$

Обозначим $A + B = U$, $\bar{A}\bar{B} = T$. Из (2) и (3) имеем $U + T = 1$, $UT = 0$.

По определению обратного элемента это означает, что $U = \bar{T}$, откуда $\bar{U} = T$, т. е. возвращаясь к исходным обозначениям $\bar{A + B} = \bar{A}\bar{B}$.

Мы вывели аксиому B_9 ; B'_9 выводится двойственным образом.

Предложение 5. Каждый из ассоциативных законов выводим из системы $S_5 = S_1 \setminus B_2, B_2$.

◀ обозначим $(A + B) + C = U$, $A + (B + C) = V$ и докажем, что $U = V$. Для этого вычислим произведения AU и AV , $\bar{A}U$ и $\bar{A}V$.

$$\begin{aligned} AU &= A((A + B) + C) \underset{B_1}{=} ((A + B) + C) A \underset{B_2}{=} (A + B) A + CA \underset{B_1}{=} \\ &= A(A + B) + AC \underset{B_2}{=} A + AC \underset{B_2}{=} A, \end{aligned}$$

$$\begin{aligned} AV &= A(A + (B + C)) \underset{B_1}{=} (A + (B + C)) A \underset{B_2}{=} AA + (B + C) A \underset{B_1}{=} \\ &= AA + A(B + C) \underset{B_4}{=} A + A(B + C) \underset{B_2}{=} A. \end{aligned}$$

Итак, $AU = A$, $AV = A$, $AU = AV$ (4).

$$\begin{aligned} \bar{A}U &= \bar{A}((A + B) + C) \underset{B_1}{=} ((A + B) + C) \bar{A} \underset{B_2}{=} (A + B) \bar{A} + C\bar{A} \underset{B_2}{=} \\ &= (A\bar{A} + B\bar{A}) + C\bar{A} \underset{B_1}{=} (0 + B\bar{A}) + C\bar{A} \underset{B_2}{=} (B\bar{A} + 0) + C\bar{A} \underset{B_2}{=} \\ &= B\bar{A} + C\bar{A} \underset{B_1}{=} \bar{A}B + \bar{A}C \underset{B_2}{=} \bar{A}(B + C), \end{aligned}$$

$$\begin{aligned} \bar{A}V &= \bar{A}(A + (B + C)) \underset{B_1}{=} (A + (B + C)) \bar{A} \underset{B_2}{=} A\bar{A} + (B + C) \bar{A} \underset{B_{10}}{=} \\ &= 0 + (B + C) \bar{A} \underset{B_{10}}{=} 0 + \bar{A}(B + C) \underset{B_2}{=} \bar{A}(B + C) + 0 \underset{B_2}{=} \bar{A}(B + C). \end{aligned}$$

Итак, $\bar{A}U = \bar{A}(B + C)$, $\bar{A}V = \bar{A}(B + C)$, $\bar{A}U = \bar{A}V$ (5).

Теперь докажем $U = V$.

$$\begin{aligned} U &= U1 \underset{B_6}{=} U(A + \bar{A}) \underset{B_1}{=} (A + \bar{A})U \underset{B_2}{=} AU + \bar{A}U \underset{(4,5)}{=} AV + \bar{A}V \underset{B_2}{=} \\ &= (A + \bar{A})V \underset{B_2}{=} 1V \underset{B_4}{=} V1 \underset{B_6}{=} V. \end{aligned}$$

Мы доказали закон B_2 , закон B_2 выводится двойственным образом ▶.

Предложение 6. Каждый из законов идемпотентности выводим из системы $S_7 = S_5 \setminus B_5, B_5$.

$$A + A \underset{B_8}{=} (A + A)1 \underset{B_{12}}{=} (A + A)(A + \bar{A}) \underset{B_1}{=} (A + A)(\bar{A} + A) \underset{B_2}{=}$$

$$= \underset{B_3}{AA} + \underset{B_4}{A} = 0 + \underset{B_1}{A} = \underset{B_4}{A} + 0 = \underset{B_4}{A}.$$

Аксиома B_4 выводится двойственным образом ►

Предложение 7. Каждый из законов поглощения выводим из системы $S_7 = S_0 \setminus B_3, B_5$;

$$\begin{aligned} \triangleleft A + AB &= \underset{B_6}{A1} + AB = \underset{B_1}{1A} + \underset{B_4}{BA} = (1 + B)A = \underset{B_1}{(B + 1)A} = \underset{B_1}{A} \\ &= \underset{B_1}{1A} = \underset{B_1}{A1} = A. \end{aligned}$$

Аксиома B_5 выводится двойственным образом ►

Теперь аксиомы придется удалять не парами, а по одной. Для обозримости выпишем систему S_7 оставшихся аксиом:

$$\begin{aligned} &Ш_3 \quad B_1 \quad B_2 \quad B_6 \quad B_7 \quad B_{10} \\ &B_1' \quad B_3' \quad B_5' \quad B_7' \quad B_{10}'. \end{aligned}$$

Предложение 8. Аксиома B_1 выводима из $Ш_3 \quad B_1' \quad B_2 \quad B_6 \quad B_{10} \quad B_1' \quad B_3' \quad B_5' \quad B_7' \quad B_{10}'$.

$$\begin{aligned} \triangleleft A + 1 &= \underset{B_9}{(A + 1)1} = \underset{B_{10}}{(A + 1)(A + \bar{A})} = \underset{B_3}{(1 + A)(A + \bar{A})} = \underset{B_3}{1\bar{A} + A} = \underset{B_1}{\bar{A}1 + A} = \underset{B_9}{A + \bar{A}} = \underset{B_{10}}{A + \bar{A}} = 1 \triangleright \end{aligned}$$

Предложение 9. Аксиома B_6 выводима из $Ш_3 \quad B_1 \quad B_2 \quad B_6 \quad B_{10} \quad B_1' \quad B_3' \quad B_5' \quad B_7' \quad B_{10}'$.

$$\begin{aligned} \triangleleft A1 &= \underset{B_{10}}{A(A + \bar{A})} = \underset{B_6}{(A + 0)(A + \bar{A})} = \underset{B_1}{(0 + A)(\bar{A} + A)} = \underset{B_3}{0\bar{A} + A} = \underset{B_3}{A + 0\bar{A}} = \underset{B_1}{A + \bar{A}0} = \underset{B_7}{A + 0} = \underset{B_6}{A}. \end{aligned}$$

Предложение 10. Аксиома B_7 выводима из системы $Ш_3 \quad B_1 \quad B_3 \quad B_5 \quad B_{10} \quad B_1' \quad B_3' \quad B_{10}'$

$$\begin{aligned} A0 &= \underset{B_6}{A0 + 0} = \underset{B_{10}}{A0 + A\bar{A}} = \underset{B_1}{0A + \bar{A}A} = \underset{B_3}{(0 + \bar{A})A} = \underset{B_7}{A(0 + \bar{A})} = \underset{B_1}{A(\bar{A} + 0)} = \underset{B_6}{A\bar{A}} = \underset{B_{10}}{0} \triangleright \end{aligned}$$

Из оставшейся системы более не удастся вывести ни одной аксиомы.

Итак, мы получили систему аксиом (назовем ее $\tilde{S}$).

$$Ш_2; \quad \bar{x}^i = x^1 + x^2 + \dots + x^{i-1} + x^{i+1} + \dots + x^k;$$

$$B_1: A + B = B + A; \quad B'_1: AB = BA;$$

$$B_3: (A + B)C = AC + BC; \quad B'_3: AB + C = (A + C)(B + C);$$

$$B_6: A + 0 = A;$$

$$B_{10}: A + \bar{A} = 1; \quad B'_{10}: A\bar{A} = 0.$$

В системе $\tilde{S}$ всего восемь аксиом, но она неудобна тем, что к B_6 нет двойственной, отчего затрудняется доказательство независимости этой системы.

Предложим более удобную и более короткую систему аксиом, из которой можно вывести все аксиомы системы $\tilde{S}$ и, следовательно, все аксиомы первоначальной системы S . Будем обозначать эту новую систему S и для отличия запишем ее малыми буквами.

Система S

$$\text{Ш}_3: \bar{x}^i = x^1 + x^2 + \dots + x^{i-1} + x^{i+1} + \dots + x^k$$

$$\delta_1: (a+b)c = bc + ac \quad \delta'_1: ab + c = (b+c)(a+c)$$

$$\delta_2: a + 0 = a \quad \delta'_2: a1 = a$$

$$\delta_3: a + \bar{a} = 1 \quad \delta'_3: a\bar{a} = 0.$$

Предложение 11. Все аксиомы системы $\tilde{S}$ выводятся из системы S .

◀ Для доказательства нужно только вывести B_1, B'_1, B_3, B'_3 :

$$a + b = \underset{\delta_2}{(a+b)} \underset{\delta_1}{1} = \underset{\delta_2}{b1} + \underset{\delta_2}{a1} = b + a \text{ (это и есть } B_1);$$

$$ab = \underset{\delta_2}{ab} + 0 = \underset{\delta_1}{(b+0)} \underset{\delta_2}{(a+0)} = ba \text{ (это и есть } B'_1).$$

Теперь, когда выведены перестановочные законы, можно поменять порядок действий в правой части аксиом δ_1 и δ'_1 и получить B_3 и B'_3 ▶.

3. *Независимость системы S .* Независимость системы S будем доказывать методом интерпретаций. Суть этого метода состоит в том, что строится структура, для которой верны все аксиомы системы кроме одной, скажем, α , которая в этой структуре не верна. Далее проводится следующее рассуждение: если бы α выводилось из остальных аксиом системы S , то в построенной структуре можно было бы вывести, что α и верна и не верна, чего быть не может, поэтому α не выводима из S .

Предложение 12. Аксиома Ш_3 независима от остальных.

◀ Рассмотрим следующую структуру: имеется n точек $x^1, x^2, \dots, x^k, \dots, x^n$ [$n > k$]; на них заданы подмножества A, B, C , а также пустое подмножество 0 и все множество 1 ; операции

определяются так: сложение = объединение, умножение = пересечение, отрицание = дополнение. Очевидно, что в этой структуре выполняются все аксиомы системы S , кроме Π_3 ▶.

В дальнейших построениях, чтобы избавиться от технических трудностей, мы положим в аксиоме Π_3 , $k=2$, так что ниже эта аксиома будет иметь вид

$$\Pi: \bar{x}^1 = x^2, \quad \bar{x}^2 = x^1.$$

Предложение 13. Аксиома δ_1 независима.

Рассмотрим структуру $\langle \{0, 1\}, +, \cdot, - \rangle$ с операциями

$$\begin{array}{c|cc} + & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 1 & 1 \end{array} \quad \begin{array}{c|cc} \cdot & 0 & 1 \\ \hline 0 & 0 & 0 \\ 1 & 0 & 1 \end{array} \quad \begin{array}{c|cc} - & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 1 & 0 \end{array}$$

Пусть $x^1 = 0$, $x^2 = 1$, $a, b, c \in \{0, 1\}$.

По определению x^1 , x^2 и по свойствам отрицания аксиома Π верна. Аксиома δ_1 верна, потому что первый столбец таблицы сложения совпадает со столбцом слева от этой таблицы, δ_1 верна, поскольку второй столбец таблицы умножения совпадает со столбцом слева от этой таблицы, δ_2 верна, ибо $0+0=0+1=1=1+1=1+0$, δ_2 верна, ибо $0 \cdot 0=0 \cdot 1=1 \cdot 1=1 \cdot 0=0$. Проверим второй дистрибутивнокоммутативный закон δ_1 : $ab+bc=(b+c)(a+c)$.

Пусть $c=0$, тогда, поскольку верна δ_2 , $ab=ba$, а это верно, так как таблица умножения симметрична. Пусть $c=1$, тогда всякое слагаемое становится равным 1 и получаем $1=1 \cdot 1=1$.

Аксиома δ_1 неверна: при $a=c=0$, $b=1$.

$$\begin{aligned} (a+b)c &= bc+ac \\ (0+1)0 &= 1 \cdot 0+00 \\ 1 \cdot 0 &= 0+1 \\ 0 &= 1 \end{aligned}$$

Предложение 14. Аксиома δ_1 независима от остальных.

▶ Рассмотрим структуру $\langle \{0, 1\}, +, \cdot, - \rangle$ с операциями

$$\begin{array}{c|cc} + & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 1 & 0 \end{array} \quad \begin{array}{c|cc} \cdot & 0 & 1 \\ \hline 0 & 0 & 0 \\ 1 & 0 & 1 \end{array} \quad \begin{array}{c|cc} - & 0 & 1 \\ \hline 0 & 0 & 1 \\ 1 & 1 & 0 \end{array}$$

Пусть $x^1=0$, $x^2=1$, $a, b, c \in \{0, 1\}$. Аксиомы Π , δ_2 , δ_2^* , δ_3 , δ_3^* проверяются по определению операций, как в доказательстве предыдущего предложения; все они верны. Аксиома δ_1 : $(a+b)c=bc+ac$ тоже верна. Пусть $c=0$, тогда $0=0+0=0$. Пусть $c=1$, тогда, поскольку верна δ_2^* , $a+b=b+a$. Это верно, ибо таблица сложения симметрична.

Однако аксиома δ_1 : $ab + c = (b + c)(a + c)$ не выполняется; при $a = c = 1$, $b = 0$ имеем $0 + 1 = (0 + 1)(1 + 1)$, $1 = 1 \cdot 0$, $1 = 0$, что неверно.

Предложение 15. Аксиома δ_2 независима от остальных.

«Рассмотрим структуру $\langle \{O, H, 1\}, +, \cdot, - \rangle$ с операциями.

+	0	H	1
0	0	1	1
H	1	1	1
1	1	1	1

·	0	H	1
0	0	0	0
H	0	1	H
1	0	1	1

$\frac{c}{\bar{c}}$	0	H	1
$\frac{c}{\bar{c}}$	H	0	0

Пусть $x^1 = 0$, $x^2 = H$, $a, b, c \in \{O, H, 1\}$. Аксиома III верна по определению отрицания, δ_2 — по таблице умножения, δ_3 — поскольку $0 + H = H + 0 = 1 + 0 = 1$, δ_3 — поскольку $0 \cdot H = H \cdot 0 = 1 \cdot 0 = 0$.

Проверим δ_1 : $(a + b)c = bc + ac$. Рассмотрим три случая: $c = 1$, $c = 0$, $c = H$. Если $c = 1$, то по δ_2 $a + b = b + a$, что верно из-за симметрии таблицы сложения. Если $c = 0$, то всякое умножение на 0 дает 0, получаем $0 = 0 + 0 = 0$. Если $c = H$, то либо $a = b = 0$, либо $a \neq 0$ или $b \neq 0$. При $a = b = 0$ имеем $0 = 0$, иначе слева получаем $1H = 1$, а справа — тоже 1, ибо хотя бы одно из слагаемых не равно 0.

Проверим δ_1 : $ab + c = (b + c)(a + c)$. Если $c \neq 0$, то все суммы равны 1, поэтому $1 = 1 \cdot 1 = 1$. Если $c = 0$, аксиома принимает вид $ab + 0 = (b + 0)(a + 0)$. Если a или b равно 0, то $a \cdot b = 0$, слева $0 + 0 = 0$. Справа будет хотя бы один сомножитель, равный 0, поэтому справа имеем тоже 0. Иначе $a \neq 0$ и $b \neq 0$, тогда во все суммы входит слагаемое, не равное 0, имеем $1 = 1 \cdot 1 = 1$.

Однако δ_2 не выполняется, что видно из таблицы сложения.

Предложение 16. Аксиома δ_2 независима от остальных.

«Рассмотрим структуру $\langle \{O, H, 1\}, +, \cdot, - \rangle$ с операциями:

+	0	H	1
0	0	0	1
H	H	0	1
1	1	1	1

·	0	H	1
0	0	0	0
H	0	0	0
1	0	0	1

$\frac{c}{\bar{c}}$	0	H	1
$\frac{c}{\bar{c}}$	1	1	H

Пусть $x^1 = 0$, $x^2 = H$, $a, b, c \in \{O, H, 1\}$. Аксиома III верна по определению отрицания, δ_2 — поскольку столбец таблицы сложения совпадает со столбцом слева от таблицы, δ_3 — поскольку $0 + 1 = H + 1 = 1 + H = 1$, δ_3 — поскольку $0 \cdot 1 = H \cdot 1 = 1 \cdot H = 0$.

Проверим δ_1 : $(a+b)c = bc + ac$. Пусть $c \neq 1$, тогда все произведения равны 0, $0 = 0 + 0 = 0$. Пусть $c = 1$. Тогда либо $a+b = 1$, либо $a+b \neq 1$. В первом случае $a = 1$ или $b = 1$, т. е. $1 = 1$, во втором $a \neq 1$ и $b = 1$, т. е. $0 = 0$.

Проверим δ_2 : $ab + c = (b+c)(a+c)$. Рассмотрим три случая: 1) $c = 1$, 2) $c = 0$, 3) $c = H$. В первом случае $1 = 1 \cdot 1$. Во втором (по δ_2) $ab = (b+0)(a+0) = ba$, что верно из-за симметрии таблицы умножения. В третьем случае $ab + H = (b+H)(a+H)$. При $ab = 1$ имеем $1 = 1 \cdot 1$, иначе $0 + H = 0$.

Аксиома δ_2' неверна, что усматривается из таблицы умножения.

Предложение 17. Каждая из аксиом δ_3 и δ_3' независима от остальных.

«Рассмотрим структуру $\langle \{0, G, H, 1\}, +, \cdot, - \rangle$ с операциями

$+$	0	G	H	1	$\cdot$	0	G	H	1	$\bar{a}$	0	G	H	1
0	0	G	H	1	0	0	0	0	0	$\bar{a}$	0	G	H	1
G	G	G	1	1	G	0	G	0	G	$\bar{a}$	0	G	H	1
H	H	1	H	1	H	0	0	H	H	$\bar{a}$	0	G	H	1
1	1	1	1	1	1	0	G	H	1	$\bar{a}$	0	H	G	1

Пусть $x^1 = c$, $x^2 = H$, $a, b, c \in \{0, G, H, 1\}$. Аксиома III выполняется по определению отрицания, δ_2 и δ_2' — потому, что первый столбец таблицы сложения и последний столбец таблицы умножения совпадают со столбцами слева от таблицы, δ_3 — поскольку $0 \cdot 0 = GH = HG = 1 \cdot 0 = 0$. Аксиомы δ_1 и δ_1' верны поскольку таблицы операций описывают объединение и пересечение четырех множеств: 0 — пустое множество, G — непустое подмножество, H — его дополнение, 1 — все множество. Эти операции удовлетворяют дистрибутивным и коммутативным законам, поэтому и аксиомам δ_1 и δ_1' . Аксиома δ_3 не выполняется $0 + \bar{0} = 0 + 0 =$

$= 0 \neq 1$. Если переопределить отрицания

$\bar{a}$	0	G	H	1
$\bar{a}$	1	H	G	1

то аксиомы III, δ_1' , δ_2 , δ_2' , δ_1 выполняются по-прежнему, δ_3 — поскольку $0 + 1 = G + H = H + G = 1 + 1 = 1$. Но δ_3' не верна: $1 \cdot 1 = 1 \cdot 1 = 1 \neq 0$.

В заключение следует отметить, что система аксиом $S \setminus \Pi_3$ является экономной, симметричной системой аксиом для булевой алгебры.

Список литературы: 1. Шабанов-Кушнаренок Ю. П. О теории интеллекта. — Проблема бинария, 1979, вып. 22, с. 3—11. 2. Шабанов-Кушнаренок Ю. П. Об алгебре конечных предикатов. — АСУ и приборы автоматики, 1979, вып. 52, с. 21—28. 3. Яглом И. М. Булева структура и ее модели. — М.: Сов. радио, 1980. — 193 с.

Поступила в редколлегию 01.02.82.

Ю. В. СОБОЛЕВ

ТЕНДЕНЦИИ РАЗВИТИЯ СОВРЕМЕННЫХ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ ПРОМЫШЛЕННЫМ ЖЕЛЕЗНОДОРОЖНЫМ ТРАНСПОРТОМ

В настоящее время значительное внимание уделяется созданию автоматизированных систем управления железнодорожным транспортом промышленных предприятий, например, таких как горнообогатительные, угледобывающие, металлургические и др. Современные АСУТП ЖДТ способствуют более четкой организации производственных процессов, обеспечивают лучшее использование сырья и технологического оборудования. С внедрением АСУТП ЖДТ повышается эффективность использования железнодорожного транспорта промышленных предприятий, что обеспечивается за счет уплотнения транспортных операций и повышения степени информированности обслуживающего персонала о состоянии перевозочного процесса.

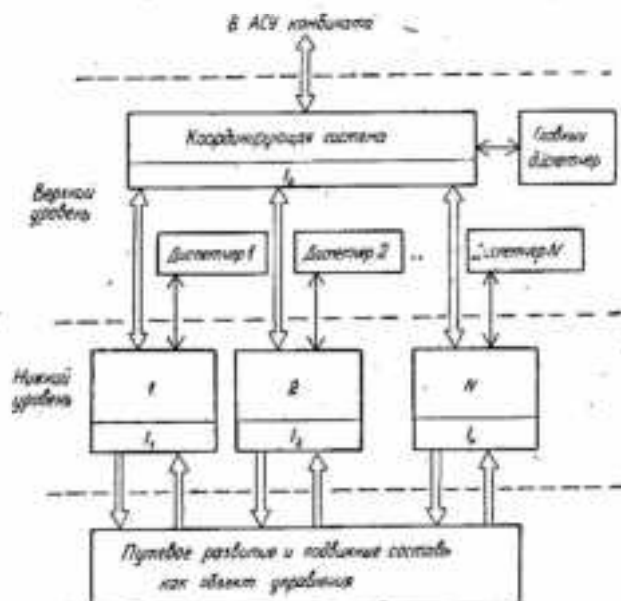
Актуальность создания АСУТП ЖДТ определяется значительной протяженностью железнодорожной сети на промышленных предприятиях.

Сутью задачи управления в АСУТП ЖДТ является формирование и реадизация оперативного графика движения составов на сравнительно небольшом интервале времени, который для промышленных предприятий составляет 30—40 мин. При создании современных АСУТП ЖДТ используются принцип децентрализованного (распределенного) управления. При этом в системе реализуется децентрализация управляющих функций и централизация диспетчерских служб, что обеспечивает системе высокую надежность и живучесть. Этот принцип основывается на иерархической структуре системы управления [1]. Перспективные АСУТП ЖДТ строятся, как правило, с использованием двух уровней иерархии.

Обобщенная структура АСУТП ЖДТ на примере горно-обогатительного комбината приведена на рисунке. Основой системы является нижний уровень, который образуют подсистемы технологических районов, включающие подсистемы станций. Последние решают задачи управления локального внутростанционного характера, относящиеся к управлению составами в пределах станции. Разделение на подсистемы технологических районов производится по технологико-топологическому признаку. Каждая такая подсистема подчиняется своему характеристическому критерию, например, $I_i = \max(p_i, \alpha_i)$, где $i=1, N$, p — вероятность безотказной работы за заданное время, α — пропускная способность составов путевым развитием технологических районов.

Подсистемы нижнего уровня могут обеспечивать работу станций в автономном режиме, что позволяет уменьшить простои и

задержки в работе железнодорожного транспорта в период пуска системы и при отказе верхнего уровня АСУТП ЖДТ. Следовательно, на нижнем уровне системы, где формируется вся информация о состоянии объекта управления (путевого развития и подвижных составов), необходимо использовать аппаратуру, обеспечивающую обязательную для устройств СЦБ безопасность функционирования. Аппаратура нижнего уровня системы должна выполнять следующие функциональные задачи: получение информации о состоянии напольных устройств железнодорожной



Обобщенная структура АСУТП ЖДТ

автоматики на перегонах, станциях и переездах (рельсовых линий, светофоров, стрелок); получение информации о подвижных составах (определение типов подвижных единиц, их числа, направления движения, скорости, ускорения); управление передвижением подвижных единиц на станциях и перегонах между ними.

Для решения указанных задач на нижнем уровне системы должны выполняться: логическая обработка информации, относящаяся к функциям маршрутно-релейной централизации (МРЦ) и управления переездами; обработка информации в соответствии с алгоритмами, обеспечивающими определение типов подвижных единиц и их учет, местоположение (координаты подвижных единиц на перегонах, станциях, в районах переездов), а также параметров их движения; передача информации между аппаратурой и объектами подсистем нижнего уровня; обмен информацией с аппаратурой верхнего уровня системы управления.

Верхний уровень АСУТП ЖДТ осуществляет контроль и координацию работы подсистем нижнего уровня, с целью оптимизации критерия качества всей АСУТП ЖДТ в целом. В качестве такого критерия может быть использовано, например, требование обеспечения минимизации времени опоздания $t_{оп}$ составов в пункт назначения $t_0 = \min t_{оп}$.

АСУТП ЖДТ обеспечивает перевозки и распределение сырья между агрегатами технологического процесса. Эту задачу обычно решают методами линейного программирования, не позволяющими учитывать динамику процесса перевозок и различные случайные возмущения в системе. С целью устранения этого недостатка на верхний уровень АСУТП ЖДТ возлагается задача динамического моделирования в реальном масштабе времени транспортных потоков комбината. Кроме этого, верхний уровень АСУТП ЖДТ обменивается необходимой информацией с АСУ комбината.

При разработке АСУТП ЖДТ используется принцип *максимальной разумной типизации проектных решений* [2]. Суть его состоит в том, что структура создаваемой АСУТП ЖДТ и ее обеспечение ориентированы на широкое применение типовых для различных промышленных предприятий одного типа, например, горно-обогатительных, черной металлургии с карьерными способами добычи руды и т. п.

Из предыдущего принципа вытекает системный подход к математическому обеспечению АСУТП ЖДТ и иерархическое построение математического обеспечения. Этот принцип состоит в том, что разрабатывается набор программных модулей, объединяемых на уровне информационных массивов. Математическое обеспечение создается с ориентацией на определенную функциональную избыточность, что дает возможность проектировать АСУ ТП с адаптацией к расширению функциональных возможностей при модернизации технологических процессов.

При технической реализации перспективных АСУТП ЖДТ основное внимание уделяется аппаратуре, используемой на нижнем уровне системы. Как и во всех АСУ, основу нижнего уровня составляют устройства для получения и передачи информации.

Наметились пути обоснованного совместного применения в единой АСУ непрерывных [3] и точечных путевых преобразователей с определением оптимальных областей использования преобразователей каждого типа. Это дает возможность осуществлять с высокой достоверностью идентификацию подвижных единиц и определять параметры их движения: координаты, направление движения, скорость и ускорение. Перспективным является создание кодовых путевых преобразователей, использующих в качестве носителя информации колеса подвижных единиц.

Для обработки информации и ее передачи начинается широкое использование микропроцессорной техники с обеспечением

у нее высоких показателей надежности, соответствующей надежности железнодорожных реле I-го класса, за счет введения аппаратной и программной избыточности. На верхнем уровне АСУТП ЖДТ используют управляющие вычислительные машины с развитым дисплейным оборудованием для отображения информации, что придает системам высокую универсальность при решении разнообразных задач. Использование указанных принципов построения и технической реализации АСУТП ЖДТ позволили повысить эффективность использования железнодорожного транспорта крупных промышленных предприятий.

Список литературы: 1. Месарович М., Мако Д., Такахара К. Теория иерархических многоуровневых систем. — М.: Мир, 1976. — 312 с. 2. Глушков В. М. Введение в АСУ. — К.: Техника, 1974. — 319 с. 3. Брылеев А. М., Котляренко Н. Ф. Электрические рельсовые цепи. — М.: Транспорт, 1970. — 256 с.

Поступила в редакцию 21.01.82.

СОДЕРЖАНИЕ

Микаев Ю. Н., Черепина И. С., Гуральник М. Д. К вопросу построения алгоритмов решения задач целочисленного линейного программирования в условиях действия возмущений	3
Путятин Е. П., Климушев В. Б. Математическая модель понятия задачи	9
Гороховатский В. А. Вопросы классификации изображений с применением алгоритмов нормализации	12
Сарниевский Н. Г., Шабанов-Кушнаренко Ю. П. Об идентификации методом сравнения динамических систем, описываемых интегралом свертки	16
Жуковин В. Е., Бурштейн Ф. В., Мусхелишвили К. Г., Стопченко Г. И. Об одном классе многокритериальных функций полезности	21
Филмонов А. Б. Применение концепции расширенной модели многосвязных объектов к задачам автономного регулирования	27
Зозуля А. Ф., Романов П. Г., Асеев Г. Г. Последовательно-декомпозиционный метод расчета балансовых уравнений сложных технологических комплексов с обратными потоками	33
Тевяшев А. Д., Гребенник В. Д. О выборе рациональной структуры модели установившегося потокораспределения в инженерной сети	41
Литвинов А. Н. Матричное представление задачи оптимального размещения отключающих устройств на инженерных сетевых системах	46
Попов В. А., Чумаченко И. В. Машинный анализ операционного и управляющего устройств на больших интегральных схемах. Ч. I	50
Сенченко Н. И., Замалеев Ю. С. Алгоритмы управления i-кратными коммутаторами для аналоговых процессоров	54
Шкиль А. С., Хаханов В. И. Трехное моделирование цифровых устройств с использованием языка кубических комплексов	59
Михайленко В. М., Литвинов А. Н., Поляков В. А., Горошков Г. А. Методологические вопросы разработки САПР-сетей коммунального хозяйства	65
Алипов Н. В., Ляшенко Ю. В. Об одном подходе к трассировке двухслойных печатных плат	71
Гвинтулян А. Х., Мазанин В. В. Модель для оценки эффективности функционирования распределенной вычислительной системы	75
Жук Н. И., Остроухов В. Д. К вопросу определения оптимальных параметров системы автоматической очистки технологического сырья	80
Подарева Н. М. Выбор достаточного спектрального состава входного сигнала для идентификации динамических систем	85
Дольберг О. М., Чесноков П. П. Определение количества оцениваемых параметров в задачах идентификации с коррелированными помехами	89
Канновский И. Ю. Исследование рекуррентных стохастических алгоритмов идентификации с постоянным шагом и негладкими функциями регрессии	91

Кольцов В. П. Решение одного класса задач методом динамического программирования	98
Петров Э. Г., Бобровский Ю. А., Бескоровайный В. В. Имитационная модель функционирования скорой медицинской помощи	101
Домрин В. Д., Шаповаленко В. А., Салыга Вад. И. Алгоритмы ортогонализации функций при идентификации технологических процессов	105
Коноваленко К. Д., Кошель Т. В. Синтез оптимальной структуры технологического процесса обработки информации о качестве по критерию	108
Гвоздинский А. Н., Дзюбенко В. Ф., Шпенев В. М. Оценка эффективности систем коммуникаций ресурсов операционной системы ЭВМ в АСУ	114
Бондаренко М. Ф. Исследование системы аксиом алгебры конечных предикатов	120
Соболев Ю. В. Тенденции развития современных автоматизированных систем управления промышленным железнодорожным транспортом	130

**АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ
И ПРИБОРЫ АВТОМАТИКИ**

Выпуск 66

Редактор *Л. Ф. Кизилов*
Художественный редактор *Т. П. Воробиевко*
Технический редактор *Л. Т. Момот*
Корректоры *Л. П. Пиленко, Л. М. Забродина*

Информ. бланк № 7004

Сдано в набор 25.06.82. Подп. и печать 13.01.83. БМ 08018.
Формат 69х90 $\frac{1}{8}$. Бумага типогр. № 3. Лит. гарн. Вяз. печать.
8,5 печ. л. 8,75 кр.-отт. 9,8 уч.-изд. л. Тираж 800 экз. Изд.
№ 1061. Зак. 2-200. Цена 1 р. 40 к.

Издательство при Харьковском государственном университете
издательского объединения «Видца школы»
310003, Харьков-3, ул. Университетская, 18

Харьковская книжная фабрика «Коммунист», 310012, Харь-
ков-12, ул. Зингедьса, 11